

情報基礎演習 2019

京都大学 国際高等教育院 喜多 一
附属図書館 北村由美
人間・環境学研究科 日置尋久
高等教育研究開発推進センター 酒井博之

Version 2019/04/01

0. まえがき

京都大学では全学共通科目として情報基礎演習が実施されています。この科目で扱う内容は学部、学科ごとに専門での必要性に応じて異なりますが、この教科書は2019年度の情報基礎演習（全学向）での利用を想定して編集されています。

本書は1章でも述べているように、大学での学習に情報通信技術（ICT）を利用するためのスキル、すなわちアカデミックな活動のための ICT スキルの獲得を主な目的としています。授業では紹介しないものも自学自習に役立てて頂くために含めています。演習も多く示していますが、実際の授業では取捨選択して取り組みます。

授業は京都大学の情報環境機構が提供する教育用コンピュータシステムの端末を用いて行うため、パーソナルコンピュータ環境はオペレーティングティングシステムとして Windows 10 を、またオフィスソフトとして Microsoft Office 2016 を想定しています。15章ではプログラミングを取り上げていますが、プログラミング言語としては初学者にも学びやすく、学術的に高度な利用も進んでいる Python を取り上げました。また、学生の皆さんの学習成果の確認のための科目内容の達成水準を示した評価基準（ループリック）を付録としてつけています。

現在、私たちが使っている情報通信技術は数多くの人々の努力により構築されているものです。このようにして技術が構築されてきたことを知ってもらい、先人の努力に感謝する意味で、コラムとして「Great Idea」を設けて紹介しています。

執筆は5・6章を北村が、15章を日置が、他の章は喜多が担当し、2017年以降の改訂にあたって酒井が参加しています。本書が皆さんの ICT スキル向上に役立てば幸いです。

なお取り上げた内容は国際高等教育院に設けられた情報科目設計ワーキングチームでまとめられた演習系科目の構成に準じて選定しています。2, 3章の執筆にあたっては企画・情報部、情報基盤課、ネットワーク管理掛の皆様に、さらに3章については、NTT コミュニケーションズ株式会社、鈴木聡介氏に、また5章の執筆にあたっては京都大学附属図書館利用支援課、原竹留美氏、櫻井待子氏、坂本拓氏に2018年度版の改訂にあたっては高等教育研究開発推進センターの岡本雅子特定助教にご協力いただきました。ご協力いただいた皆様に心より御礼申し上げます。

授業を担当される教員の皆様へ

大学設置基準では演習科目については授業時間外の学習も含めて 45 時間の学修¹で 1 単位を発給することと想定されています。本書は 2 単位が発給される演習科目向けに、授業時間外での学習での利用も考慮して編集しています。本書で紹介している内容をすべて授業で取り上げることは難しいと思います。一部を予習や復習、課題の際に受講者に参考にしてもらうなどの活用をご検討ください。

アカデミックな活動に求められる ICT スキルは学年の進行とともに高度になりますし、新しい技術やサービス、それに伴う適正な利用のための注意事項などについて学生が自ら学習する能力を獲得することが求められます。本書でも第 1 章で「**ICT スキルを自主的・継続的に獲得する自学自習能力を身につけること**」を求めています。授業で紹介しない内容について自学自習することは、この点からも重要であると考えています。

本書は CC-BY-NC-ND ライセンスによって許諾されています。ライセンスの内容を知りたい方は <https://creativecommons.org/licenses/by-nc-nd/4.0/deed.ja> でご確認ください。



¹一般的に用いられる「学習」とは異なり、授業時間以外に主体的に行われる予復習の学びを含み込む用語として「学修」が用いられます。

目次

0. まえがき	2
授業を担当される教員の皆様へ	3
1. この授業の目的, 評価基準, 進め方	10
1.1 この授業の3つの目的	10
1.2 教科書で扱う内容の位置づけ	12
1.3 評価基準	13
1.4 授業の進め方	13
1.5 「効果」と「効率」からの評価と改善	14
1.6 この教科書の構成	14
2. コンピュータの基礎	15
2.1 学習の目的	15
2.2 情報通信技術の進展	15
2.3 パーソナルコンピュータの構成	18
Great Idea! ストアードプログラム方式	20
2.4 オペレーティングシステムの役割	22
Great Idea! オペレーティングシステム	23
2.5 ファイルとファイルシステム	26
2.6 OS とユーザインターフェイス	37
Great Idea! Window, Icon, Menu & Pointer	37
2.7 自立したパソコン利用者に求められるスキル	38
2.8 キーボードのタイピング	43
Great Idea! かな漢字変換	48
2.9 ソフトウェアのライセンスと保守	48
参考文献	53
3. ネットワークの基礎	54
3.1 学習の目的	54
3.2 仮想社会の安全運転	54
3.3 情報セキュリティ e-Learning の受講	55
3.4 ネットワーク上の情報システム	55
3.5 ローカルエリアネットワーク (LAN) の基礎	57
3.6 インターネットの基礎	64

Great Idea! インターネットプロトコル	66
Great Idea! 公開鍵暗号	75
3.7 パーソナルコンピュータのネットワーク接続	77
3.8 電子メールの利用	78
3.9 デスクトップの仮想化	81
3.10 京都大学のネットワークサービス	82
3.11 情報通信技術 (ICT) の利用とリスク	84
4. 知的生産について	87
4.1 本章の目的	87
4.2 知的生産とは	87
4.3 学園祭の活動と知的生産	88
4.4 大学生活と知的生産	89
4.5 知的生産を構成する活動	90
4.6 知的生産と ICT 活用	91
4.7 知的生産スキルの獲得と学習の方略	92
参考文献	92
5. 学術情報の探索	93
5.1 本章の目的	93
5.2 学術情報と図書館	93
5.3 検索を行う前に	96
5.4 参考文献リストの読み方・書き方	99
5.5 検索の実際	102
5.6 困ったら図書館に	102
参考文献	102
Great Idea! ガーフィールド博士と引用文献データベース	103
6. 文献管理ツールを活用しよう	104
6.1 学習の目的	104
6.2 文献管理ツールとは?	104
6.3 RefWorks を使ってみる	105
7. 表計算ソフトによるデータ処理 (1)	117
7.1 学習の目的	117
7.2 表計算ソフトの利用目的	117

7.3	表計算ソフトの基本構造	118
	Great Idea! 表計算ソフトの開発	120
7.4	データの入力方法	121
7.5	データの並べ替え	127
7.6	データの集計	128
7.7	データのクロス集計	130
8.	表計算ソフトによるデータ処理（2）	134
8.1	学習の目的	134
8.2	シミュレーションとは	134
8.3	動的な状況のシミュレーションの構成要素	134
8.4	借入金の返済のシミュレーション	134
8.5	バネ・ダンパモデルのシミュレーション	137
9.	表計算ソフトによるデータ処理（3）	142
9.1	学習の目的	142
9.2	データをグラフで表示することの意味	142
9.3	データをグラフで表現する時の注意事項	142
9.4	グラフを Excel で作成する際の用語	143
9.5	散布図を Excel で描く	144
9.6	棒グラフや折れ線グラフを Excel で描く	149
9.7	ヒストグラムを描く	150
10.	表計算ソフトによるデータ処理（4）	153
10.1	学習の目的	153
10.2	対数グラフとは	153
10.3	Excel での対数グラフの作成	156
10.4	グラフにおける単位の表記	160
11.	構造化された文書の作成	162
11.1	学習の目的	162
11.2	論文やレポートの構造	162
11.3	人がすること，コンピュータに任せること	163
11.4	スタイルとテンプレート	165
11.5	パラグラフの書き方	168
11.6	章立ての見出しの書き方	169

11.7	箇条書き・脚注・ページ設定	170
11.8	アウトライン表示の活用	172
11.9	参考文献の記載方法とその引用	174
11.10	文章の校閲方法	176
11.11	PDF 形式の文書の作成	177
	参考文献	178
12.	文章への図表の挿入	179
12.1	学習の目的	179
12.2	Word の文章への図表の挿入	179
12.3	学術的文章での図表番号，タイトル，説明の付け方	181
12.4	Word での図表番号の付与と相互参照	181
12.5	Word と Excel, PowerPoint の連携	184
12.6	PowerPoint の図の挿入	185
12.7	Excel の表やグラフの挿入	186
13.	文書での数式の扱い	187
13.1	学習の目的	187
13.2	表現の手段としての数式	187
13.3	変数は傾いている	187
13.4	物理量と単位の表記	188
13.5	Word での数式の書き方	189
13.6	2次元形式と行形式	189
13.7	文中数式と独立数式	190
13.8	式番号と相互参照	190
	参考文献	193
14.	プレゼンテーションについて考えよう	194
14.1	学習の目的	194
14.2	プレゼンテーションの目的と聞き手	194
14.3	プレゼンテーションでの留意事項	194
14.4	スライド作成の基本	195
14.5	発表の作法	201
14.6	グループでのプレゼンテーション	203
14.7	聞き手としての態度	203

14.8	プレゼンテーションの評価	203
14.9	ポスター発表について	204
	参考文献	206
15.	プログラミングの基礎	207
15.1	学習の準備	207
15.2	学習の目的	207
15.3	プログラムとプログラミングの概要	208
	Great Idea! 自動プログラミング	217
15.4	変数によるモデル化	218
15.5	制御構造 — 条件に基づく処理	222
15.6	プログラミング演習の準備	225
	参考文献	234
16.	あとがき	235
	付録 略語集	236
	別表 1 評価基準	244

1. この授業の目的，評価基準，進め方

1.1 この授業の3つの目的

大学での学習では文献などの情報探索を行ったり，統計データを活用したりしつつレポート作成し，プレゼンテーションを行うなど様々な場面で ICT の活用が求められます。すなわち**アカデミックな活動に必要な ICT スキルを身に着けること**が求められます。また，大学生になると，クラブやサークル，アルバイトや NPO での活動など授業外の活動も拡大し，より主体的に活動を実施するようになります。このような活動においてもフォーマルな場面で ICT を活用するスキルが求められます。これに必要なスキルはアカデミックな活動のための ICT スキルと近いものです。この授業の第一の目的は**アカデミックな活動に必要な ICT スキル**を身に着けることです。

次のポイントは大学に入学して一人暮らしを始める学生も多くなりましすし，ネットワークに接続されたパーソナルコンピュータ（PC）を個人で占有利用するようになることです。このため PC を適切に運用することなど，**自立した ICT ユーザとしてのスキル**も必要になります。これがこの授業の第二の目的です。

大学でのアカデミックな活動は学年進行とともに内容が高度化，専門化し，これに伴ってより高いレベルでのスキルが求められます。これを1年生の限られた時間で学習することは難しいですし，みなさん自身にとっても必ずしも利用するとは限らない高度な内容を学習することの意義を見いだせないかもしれません。しかしながら，2年生以降，まとまって ICT スキルを学ぶ授業などが設けられているとは限りませんので自身で主体的に学ぶことが求められます。この授業の第三の目的は，学年進行に伴って高度化する学術的な活動のための **ICT スキルを自主的・継続的に獲得するための自学自習能力**を身に着けることです。

これらのことをまとめるとこの授業の目的は以下ようになります。

- **アカデミックな活動に必要な ICT スキルを獲得すること**: 大学が提供するさまざまな情報サービスを適切に利用できるようになるとともに，アカデミックな活動に必要な情報探索，データの処理，レポート作成とプレゼンテーションなどのための ICT スキルを獲得することです。
- **自立した ICT ユーザとなること**: パーソナルコンピュータやネットワークを自立したユーザとして適切に利用し管理する能力を身に着けることです。
- **ICT スキルを自主的・継続的に獲得する自学自習能力を身につけること**: この授業では扱いませんが，学年が進行し，より専門的な能力を身に着けて行く上で必要なスキルを主体的に獲得することです。

科目は半期1コマで1年生の最初の学期で実施されることを想定しています。これら全体を通して学習の目的を表現するならば図2のように大学での学習の生産性を早い段階で高め，大学での学習をより成功させるための ICT スキルの獲得と言えるでしょう。

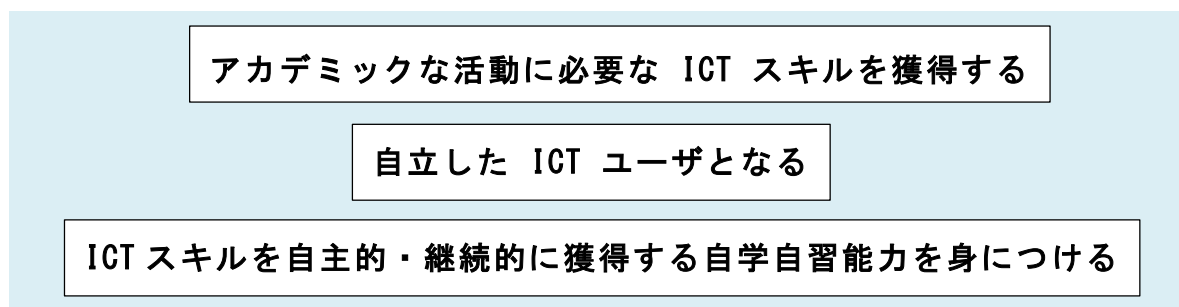


図1 この科目の3つの目的

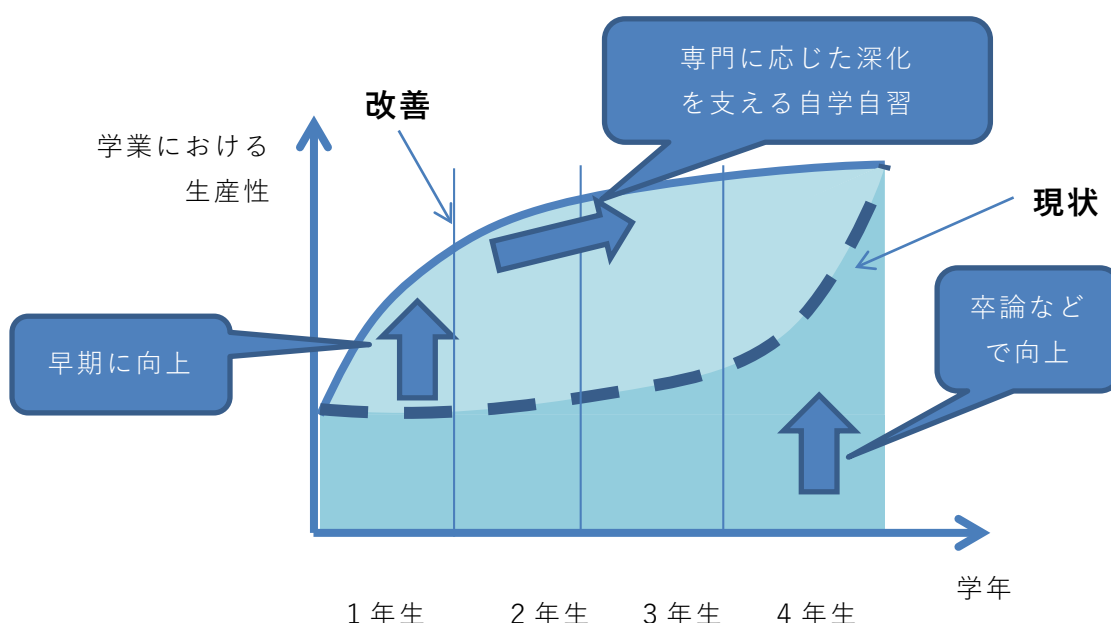


図 2 学業における生産性の向上

1.2 教科書で扱う内容の位置づけ

さて、まえがきでも説明したように、この教科書ではコンピュータのオペレーティングシステムとして Windows 10 を前提としています。また教科書で扱う主なトピックのうち、文書作成、表計算、プレゼンテーションにおいては、演習用のツールとして Microsoft Office 2016 の Word, Excel, Powerpoint を取り上げています。その他に文献管理では Refworks を取り上げ、プログラミングについては Python(Anaconda)を紹介しています。これらはいずれも京都大学の教育用コンピュータシステムで利用可能で、演習でも使うことになるでしょうし、また一般にもよく利用されています。

しかしこの教科書で扱っているツールに習熟すれば、大学のアカデミックな活動での ICT 活用シーンのすべてに対応できるとは限りません。アカデミックな活動に利用されるツールは他にも数多くあります。すでに説明したように、大学で利用されるツールは、学年が上がるにつれ専門化し、みなさんが今後専門とする学問分野あるいは所属する研究室によっては、この教科書では紹介していないツールを扱う必要も出てくるでしょう。

またこの教科書で取り挙げているツールについても、同じ用途でオープンソースソフトウェアとして無償で利用できるものや、ネットワークを通

じてクラウドサービスとして使うものなど他のツールもあります。さまざまなツールを知れば，自分の目的により適したもの，自分にとってより使いやすいものを積極的に選択することが可能になります。そのような場面で ICT スキルを自主的・継続的に獲得する自学自習能力が重要になります。

この授業では，単に演習で扱うソフトウェアの使い方を学ぶのではありません。コンピュータを使ってアカデミックな活動の生産性を高めるためには，自分が使っているシステムはどんな仕組みのものなのかを把握した上で，さまざまな活動において，どのように情報をとらえて，どのように処理し，どのように表現すればよいのかといった ICT 利活用における本質的な考え方を学ぶことが重要です。このようなことを意識しておくことで，授業での学習を通じて，ICT スキルを自主的・継続的に獲得する自学自習能力として，汎用的で長期的に有用で，さまざまな場面に応用しうる基盤的な力を身につけられることでしょう。

1.3 評価基準

それでは，どのような能力を身につければいいのでしょうか。「別表 1 評価基準」は学習する事項を縦軸に，達成すべきレベルを横軸に取り，各マス目にその事項の求められる達成レベルを示したものです。レベル C は最低限，達成して欲しいレベル，レベル A はこの授業で（あるいは 1 年生で）目標とすべきレベル，レベル S は卒業時までには身につけてほしいレベルとして設けています。

演習 1. ICT スキルの自己評価

別表 1 の各項目について，ご自身の現在のスキルレベルを確認し，達成すべき目標との差を理解してください。

1.4 授業の進め方

大学での授業は授業時間内と授業時間外の学習で構成されます。授業の進め方は担当教員によって異なりますが，本書では演習課題の指示に沿って各自で演習を進め，課題を授業支援システム PandA で提出する形で進

めることを想定しています。授業中に終了しなかった課題は授業後に自習し PandA で提出することも想定しています。このほか，授業時間外に行う学習課題も設定されますが，これも PandA で提出することが行われるでしょう。

1.5 「効果」と「効率」からの評価と改善

スキルを身に着けて行く上での視点として「効果」と「効率」という2つの視点をもって自身の活動を自分で評価してください。

- **効果的 (Effective)**とは実現したい目的により適した方法が取られていて，よりよい成果が得られていることを言います。これに対して
- **効率的 (Efficient)**とは同じことを実現するのにより早く，より正確に，より少ない手間で行える方法を用いていることを言います。

ご自身の活動をこの2つの視点で常に自己評価し，より効果的で効率的に活動できるように

- 継続的に学習し作業を改善すること
- を心がけてください。大量の情報を高速で扱えるコンピュータの特性から，PC など情報機器やサービスの利用ではこのような継続的な学習と改善が極めて重要です。

1.6 この教科書の構成

この教科書は授業で取り組む内容について，学習の目標を示し，学習する内容について解説します。演習問題をいくつか設定していますが授業で実施する演習課題については別途，具体的に指示されます。

2. コンピュータの基礎

2.1 学習の目的

現代社会ではコンピュータは「パーソナルコンピュータ」だけでなく、名前こそコンピュータとは呼びませんが、スマートフォンやタブレット端末もパーソナルコンピュータと似たような構成を持つコンピュータです。スーパーコンピュータや Web サーバなどもコンピュータですし、身の回りの家電製品やゲーム機、自動車などにもコンピュータが組み込まれています。

この章では大学生になって利用が本格化するパーソナルコンピュータの自立した利用者となるため、

- 自立したパーソナルコンピュータの利用者に求められるスキル

について説明したあと、これを体系的に理解するために

- パーソナルコンピュータの構成
- オペレーティングシステムの役割

について学びます。

また、パーソナルコンピュータの操作の中でも、アカデミックな活動をより効率良く行うための基礎的なスキルとして

- キーボードのタイピング

について学びます。

2.2 情報通信技術の進展

情報通信技術がどのように進展してきたかの概要を知ることは今後、どのようなことが情報通信技術で可能になるかを想定し、どのような利用を考えて自身のスキルを身に着けるかを考えるヒントになります。

2.2.1 コンピュータの誕生

電子技術を用いて情報を高速に処理しているのが現代のコンピュータ

です。このようなことが始まったのは 1940 年前後からで、大量の数値計算を必要とする科学領域などで利用されました。

当初は真空管という白熱電球に似た仕組みを利用したため、大きくて寿命や信頼性も低いことが問題でしたが、トランジスタなど半導体を用いることで寿命の問題が解消され利用が広がりました。1960 年代にはメインフレームと呼ばれる大型のコンピュータが企業での事務処理などで利用されるようになりました。1970 年代には、個人がコンピュータを専有してさまざまな用途に利用するパーソナルコンピュータという利用形態が立ち上がります（図 3 参照）。



図 3 コンピュータの発展

* <https://commons.wikimedia.org/wiki/File:Eniac.jpg>

** https://commons.wikimedia.org/wiki/File:Bundesarchiv_B_145_Bild-F038812-0014,_Wolfsburg,_VW_Autowerk.jpg

*** https://commons.wikimedia.org/wiki/File:Xerox_Alto_mit_Rechner.JPG

**** https://commons.wikimedia.org/wiki/File:Apple_II_transparent_800.png (2017/1/3 アクセス)

2.2.2 半導体技術の進展

コンピュータの利用が拡大した背景にはコンピュータを構成する複雑な回路を 1 つの半導体チップ上に構成する集積回路技術と集積回路としてコンピュータを半導体チップ上に構成したマイクロプロセッサが出現したことがあります。これによりコンピュータを大量に安価に製造することが可能になりました。集積回路技術はその後、急速に発展し、チップ上に構成するトランジスタの数も個々のトランジスタの動作速度もその性

能が何桁も変わる技術革新が今も続いています（図 4 参照）。

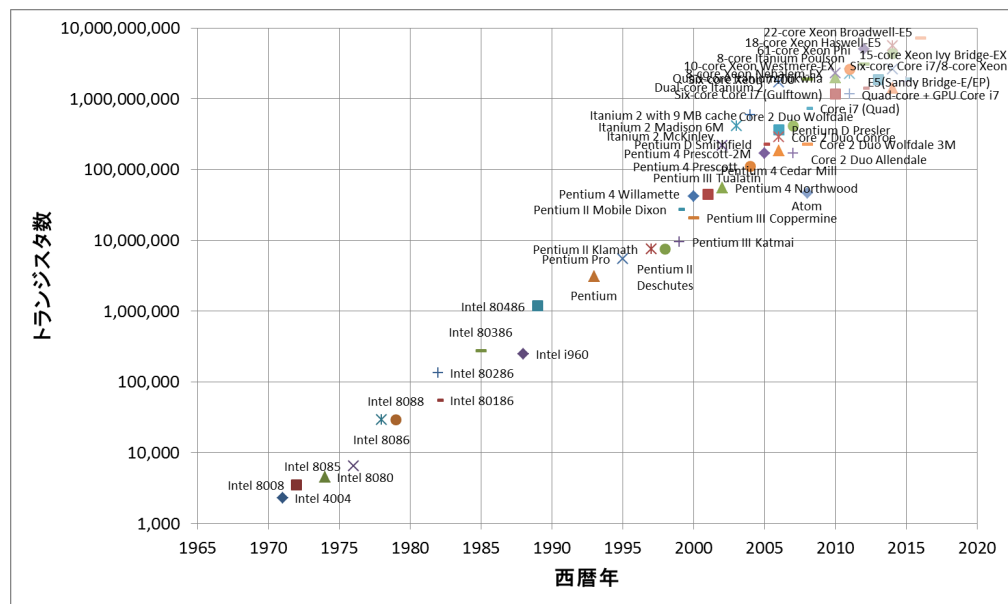


図 4 マイクロプロセッサ上のトランジスタ数

https://en.wikipedia.org/wiki/Transistor_count#Microprocessors

(2017 年 1 月 2 日アクセス) から Intel 社製のプロセッサを抽出してプロット

2.2.3 インターネット

1970 年代からのもう一つの重要な進展はインターネットです。コンピュータどうしをデジタル通信で接続したネットワークを構成して情報の処理と通信が一体化されました。1990 年代にはインターネットが商用にも利用可能になり、パーソナルコンピュータの高度化とも相まって急速に発展普及します。そこでは電子メールや World Wide Web などの情報サービスが展開され、現在では World Wide Web を基盤としてさまざまな情報サービスが提供されるようになりました。さらに、1990 年代から携帯電話網を介して携帯端末がインターネットにつながるようになり、その利用がさらに拡大します。持ち歩くことができ常時インターネットに接続され、使い勝手もよいスマートフォンの出現はパーソナルコンピュータとは異なった情報通信技術の利用を可能にしています。

2.2.4 メディアのデジタル化

コンピュータやデジタル通信の性能向上に伴い、それまで個別に扱われていた文書、写真、音声、動画などのさまざまなメディアがデジタル化され、一元的に扱えるようになってきたこともコンピュータの利用を拡大しています。さらに、コンピュータを用いることにより、利用者の操作に応じて動くコンピュータを用いたゲームが出現したり、インターネットを介してさまざまなサイトで連携して情報を提供する「ハイパーメディア」としての WWW が出現したりしています。

2.2.5 情報技術と社会

他方で、このような世界規模でのコンピュータネットワークの出現はさまざまな社会的問題も引き起こしており、情報通信技術を適正に利用するために情報セキュリティ対策や倫理的な行動が求められています。

2.3 パーソナルコンピュータの構成

2.3.1 コンピュータの構成

コンピュータはハードウェア（電子回路などの物理的装置）とソフトウェア（プログラム）で構成されています。

1) コンピュータのハードウェア構成

パーソナルコンピュータは図 5 のような装置で構成されています。

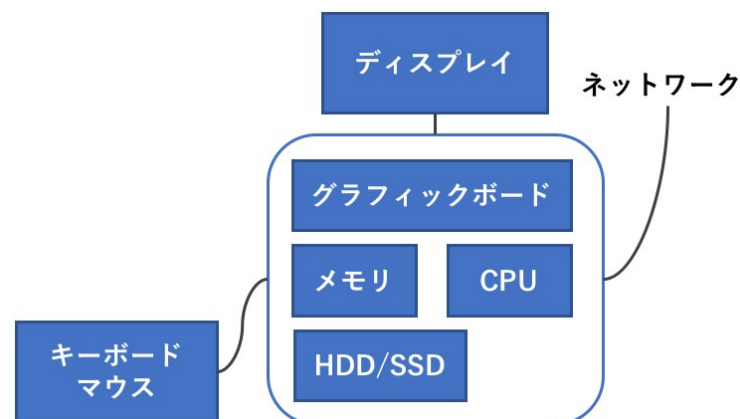


図 5 コンピュータの構成（ハードウェア）

- **CPU (中央処理装置, Central Processing Unit)** : 主記憶からプログラムを1ステップずつ読み出し, その命令に応じて, 主記憶上のデータや CPU 内のレジスタと呼ばれる記憶装置上のデータに演算を加えたり, 転送したり, データに応じて次に実行するプログラム切り替えたりするコンピュータの中核的な装置です。
- **主記憶 (メモリ)** : CPU が実行するプログラムや操作するデータを格納する装置です。高速な読み出しや書き込みが求められるので半導体で構成されたメモリが用いられますが電源を切るとその内容が失われます (揮発性メモリと呼ばれます)。
- **二次記憶装置 (ストレージ)** : 大量のプログラムやデータを永続的に保持するための記憶装置です。電源が切れてもデータが保存される磁気ディスク, フラッシュメモリ, CD や DVD ドライブなどの光学的な記憶装置が用いられます。
- **入出力装置** : 情報処理のための入力を外部から得たり, 結果を提示したりするための装置です。グラフィックボード, ディスプレイ, キーボード, マウス, タッチパネル, プリンタなどが広く用いられています。
- **ネットワーク回線** : インターネットなどコンピュータどうしを接続するためのネットワーク回線です。多くのパーソナルコンピュータでは有線や無線の LAN に接続できるようになっています。

2) コンピュータの動作

コンピュータの基本動作は以下のように極めて単純です。

- 何らかの手段でメモリ上にプログラムや必要なデータを配置します。
- CPU にプログラムの実行開始場所を与えます。
- あとはコンピュータ自身が以下の動作を繰り返します
 1. CPU はメモリ上のプログラムから1ステップ分を読み出し, 指示に従って計算, データの転送などを行います。
 - 計算結果をメモリに書き出す場合もあります。
 - 入出力を行う場合もあります。
 2. CPU は実行の対象をプログラムの次のステップに進めます。
 - 実行場所はプログラムによって変化する場合があります。

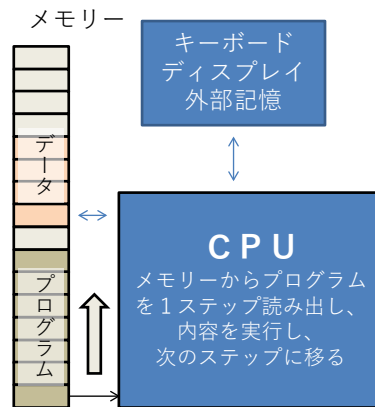


図 6 コンピュータの動作（ハードウェア）

Great Idea！ ストアードプログラム方式

このようなコンピュータの構成は**ストアードプログラム方式**と呼ばれています。複雑な機能の実現をプログラムに委ねることにより、プログラムを変えるだけで多様な利用を実現しています。家電品に組み込まれている小さな制御用のコンピュータからスーパーコンピュータまで共通の構成方法です！

2.3.2 ソフトウェア

ソフトウェアとはコンピュータ上で実行するプログラムのことです。ストアードプログラム方式の採用により、ハードウェアを変えることなく、メモリ上に展開するプログラムを利用したい用途ごとに用意することでコンピュータを様々な用途に使うことができます。その後の技術進歩により、ハードウェアは単純な動作を極めて高速に実現するようになりました。これに対して、用途ごとの複雑な動作はソフトウェアで実現しています。このため高度なソフトウェアではプログラムの行数は数億行にも達しています。

パーソナルコンピュータではソフトウェアは効果的で効率的な運用のためいくつかの形態が取られています。

- **アプリケーションソフト**（例えば Word, Excel）：利用者の具体的なニーズを実現するソフトウェアです。

- **オペレーティングシステム(OS, 例えば Windows)：**ハードウェアとアプリケーションソフトウェアの間に介在して，アプリケーションソフトの実行環境を整えます。

またオペレーティングシステムを補助したり，複数のアプリケーションソフトで共通に利用できるようにしたりする補助的なソフトウェアとして以下のようなものがあります。

- **ミドルウェア：**複数のアプリケーションソフトで共用できる機能（ライブラリ）を実装した補助ソフトウェアです。
- **デバイスドライバ：**プリンタなど機種に応じて装置を駆動する機能をオペレーティングシステムに提供するソフトウェアです。

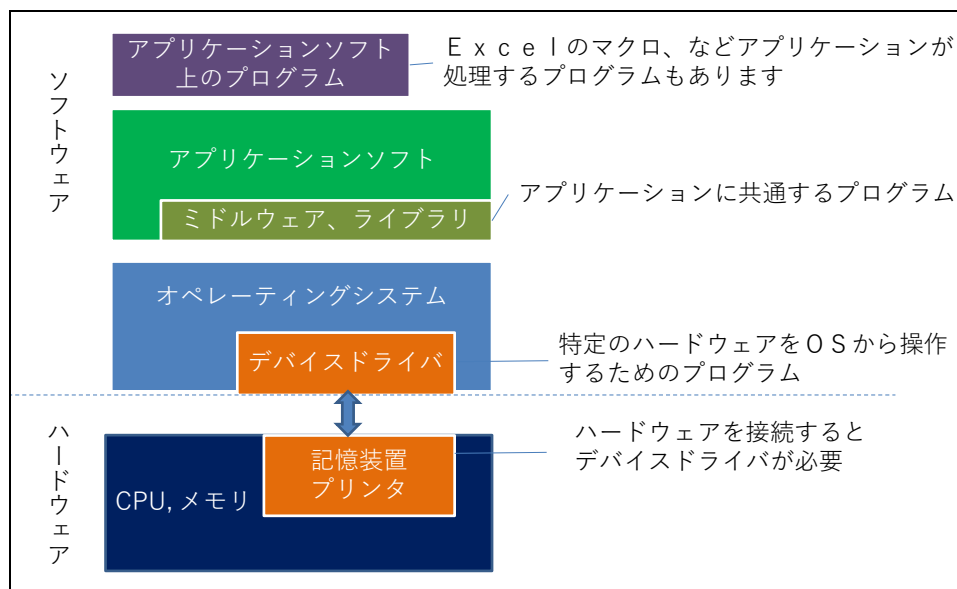


図 7 ソフトウェアの構成

2.4 オペレーティングシステムの役割

2.4.1 OS について学ぶことの意義

オペレーティングシステム（Operating System, OS）について学ぶことにはアプリケーションソフトの操作ほど直接のありがたさはありませんが、

- コンピュータ全般の操作効率が上がります
- コンピュータを適切に管理できるようになります
- コンピュータに生じた問題を把握しやすくなります

遠回りのようですが、OS について学ぶことでコンピュータを使った作業の生産性が上がります。

2.4.2 OS の役割

OS はパーソナルコンピュータの動作に関して以下のようなさまざまな役割を担っています。

- アプリケーションソフトの起動、停止、メモリと CPU による実行の割り当て
- ファイルの作成、移動、消去
- アプリケーションソフトのインストール（導入）や削除
- ネットワークでの通信
- キーボードやマウスからの入力
- プリンタへの出力
- ユーザアカウントの作成、ユーザの認証と権限の管理
- ウィンドウの描画、コピーペースト

具体的には図 8 のようなことを担っています。

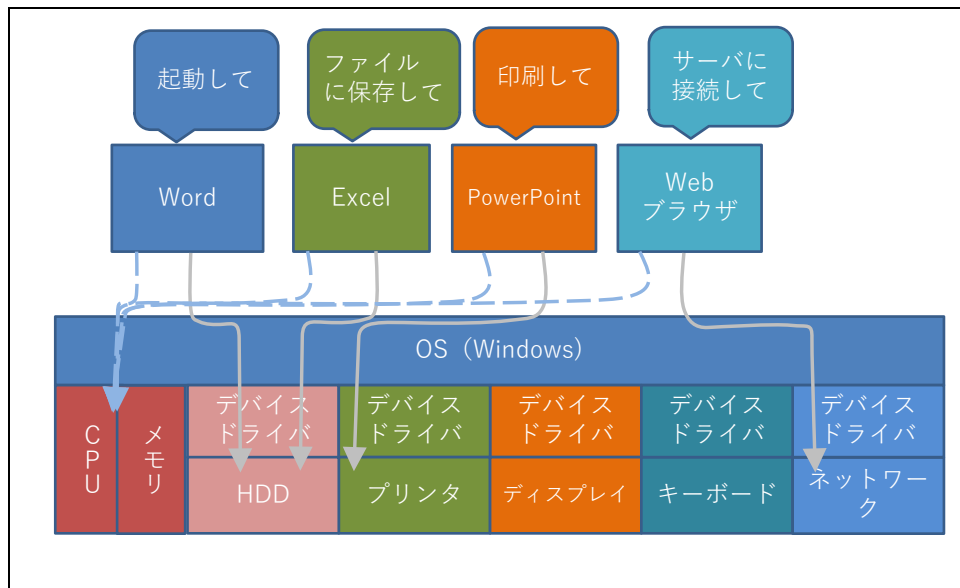


図 8 オペレーティングシステムの役割

Great Idea ! オペレーティングシステム

初期のコンピュータでは、ハードウェアごとに個々にアプリケーションソフトが作られていました。現代ではOSの存在により、開発者にとっては

- ハードウェアもアプリケーションソフトもOSだけを意識して開発すればよくなっています。
- アプリケーションソフト側ではウィンドウの操作などを詳細に記述する必要がなくなります。

ユーザにとっては

- 個々のアプリケーションソフトを同様の操作で使えます。

という利点があります。

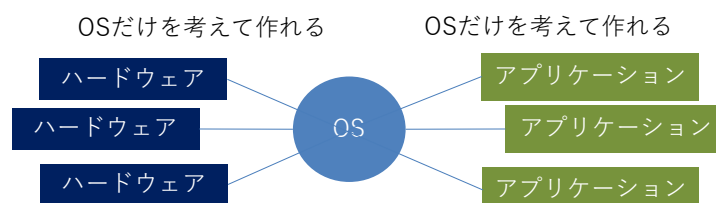


図 9 仲介役としてのOS

2.4.3 動作から見たソフトウェアの動き

次にソフトウェアの動きを見て行きましょう。

パーソナルコンピュータを起動すると、

1. まずコンピュータに固定的に設定されているプログラム（ファームウェア）が起動されます。
2. このファームウェアが接続されているディスクドライブなどから、ドライブ上に収納されているオペレーティングシステム(OS)を読み込むためにブートローダを読み込み、これに実行を移します。
3. ブートローダがさらにドライブ上の OS を読み込み、これに実行を移します。
4. OS が起動し、さまざまな準備が終わるとユーザはウィンドウの操作など OS を介してアプリケーションソフトを起動して利用します。

現代の OS はさまざまな作業を実行しています。パーソナルコンピュータを停止するためには、**シャットダウン**の操作を行ってこれらを適正に終了させた上で電源を切ることが求められます。

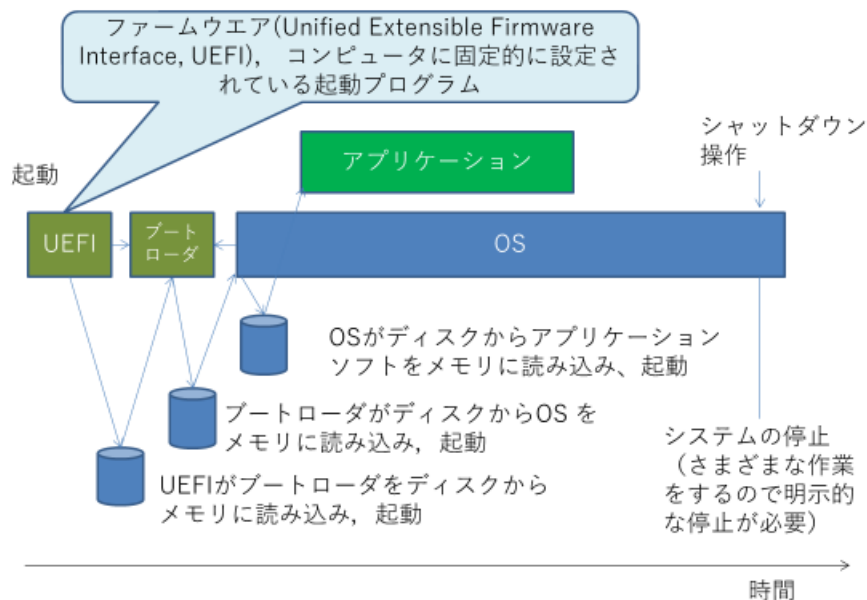


図 10 動作からみたソフトウェアの動き

2.4.4 プログラムの並行実行と CPU とメモリの利用

パーソナルコンピュータではさまざまなアプリケーションソフトを並行して実行することができます。このために、OS はそれぞれのアプリケーションソフトと必要なデータ用の領域をメモリ上に配置します。実行にあたっては、それぞれのプログラムの CPU での実行を短い間隔で切り替えて、あたかも複数のプログラムが同時に実行されているようにしています。

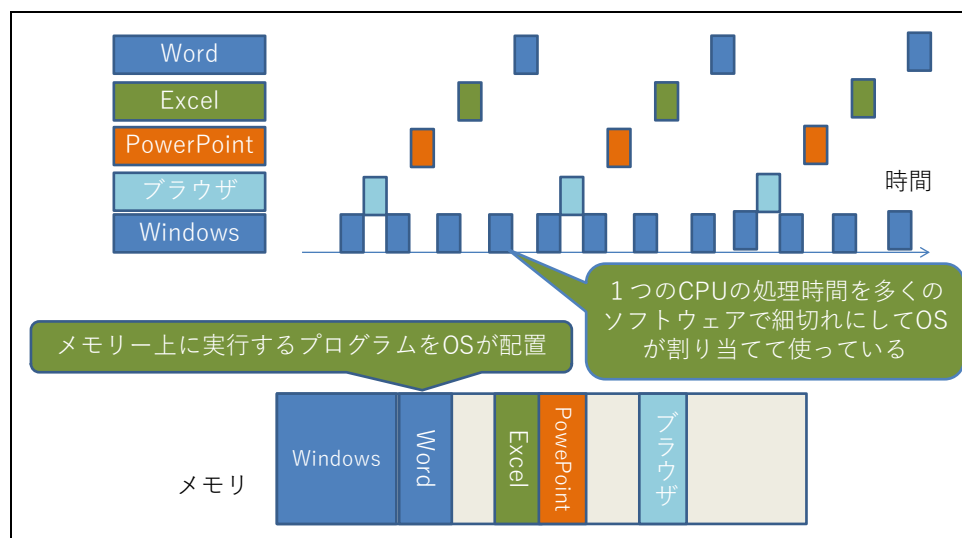


図 11 プログラムの並行実行

2.5 ファイルとファイルシステム

2.5.1 ファイル

コンピュータで利用する文書、画像、音声、プログラム（ソフトウェア）などは、ハードディスクなどの記憶装置上に「ファイル(file)」として記録されます。

2.5.2 ファイルの容量

ファイルに記録されているデータの量やディスクなどの記憶装置の容量、OS や後述のファイルシステムによって制限されるファイルの最大サイズなどは「バイト(byte)」を単位として示されます。1 バイトは2 進数 8 桁（8 ビット）で、後述の文字コードでは ASCII コードなどで1 文字を格納することに適したサイズです。

大量のデータ量を表すために、これに k(キロ), M(メガ), G(ギガ), T(テラ)などの補助単位をつけて用いられます。例えば 1Mbyte とは 10^6 byte であり、英数字 100 万文字程度のサイズということになります。²

表 1 補助単位

記号	呼び方	意味
k	キロ	10^3
M	メガ	10^6
G	ギガ	10^9
T	テラ	10^{12}
P	ペタ	10^{15}

² コンピュータでは 2 進数が主に用いられることと、 $2^{10} = 1024 \div 1000 = 10^3$ であることから 1024 バイトを 1kbyte とすることもあります。

2.5.3 ファイルフォーマットと拡張子

1) ファイルのフォーマット

コンピュータではファイルとして文書ファイル、画像、音声、動画、プログラムなど、さまざまなデータを格納します。それぞれをプログラムで適正に操作するために、データをどのような形式でファイルに格納するかを定めたものがフォーマットです。

同じ画像でもさまざまなフォーマットが用いられており、データを処理するプログラムが、そのフォーマットのファイルを取り扱えるかどうか重要です。特にデータを作成した機器やソフトウェアとそのデータを読みこむ機器やソフトウェアが異なる時には注意しなければなりません。

- 文書ファイル

- テキストファイル: 文字と改行などいくつかの制御用の符号で構成されたファイルです。文書としての書式情報は含まれません。電子メールのメッセージや Web ページを記述する HTML 形式のファイルは書式の情報なども含まれますが、すべて文字データとして読み書きできる文字で書かれていて、その意味ではテキストファイルの形式をとっています。
- ワードプロセッサ用の文書ファイル: 例えば Word の文書ファイルです。ソフトウェアやバージョンによって異なるフォーマットが用いられるため、異なるワードプロセッサソフトを用いる人に文書ファイルを渡す際には利用可能なフォーマットを選ぶ必要があります。
- 印刷用フォーマット: 印刷した際のイメージに近い形式のフォーマットで **PDF (Portable Document Format)** 形式が広く用いられています。このほか EPUB など電子書籍用のフォーマットもあります。

- 表計算ソフト用のデータ

- 表計算ソフト用フォーマット: Excel など表計算ソフトごとにソフトウェアが定めたフォーマットが用いられます。運用に当たってはワードプロセッサの文書ファイルと同様の配慮が必要です。

- **CSV 形式フォーマット**: CSV とは Comma Separated Values の略で表の 1 行を「セル」の文字列や数値をカンマ(,)で区切って 1 行で書いたテキストファイルの形式です。
- **画像ファイル**
 - **ベクター形式のデータ**: 直線や曲線など図を構成する要素の幾何学的な情報で構成されたデータ。ドロー系のソフトウェアのデータを保存するファイルのほか、SVG 形式などデータを共通利用するためのフォーマットがあります。
 - **ラスタ形式のデータ**: 写真やスキャナで読み込んだ画像などは格子状の点（画素）の色，明るさのデータで表現されています。このような形式をラスタ形式と呼びます。**JPEG, TIFF, PNG, BMP, GIF** などさまざまなフォーマットが用いられています。データ量を少なくするため後述の非可逆圧縮がしばしば用いられます。
- **音声ファイル**: WAV, AIFF, MP3 などさまざまなフォーマットがあります。MIDI は電子楽器の演奏用のフォーマットで音声そのもののフォーマットではありません。
- **動画ファイル**: AVI, MP4, MOV, FLV などさまざまなフォーマットがあります。
- **プログラム**: そのまま実行できるコンピュータのプログラムのためのフォーマットです。Windows では Portable Executable (PE) 形式のフォーマットが用いられ，拡張子 .exe が付けられます。そのままコンピュータで実行可能なファイルですので，マルウェアとしての侵入などに注意が必要です。
- **アーカイブファイル**: 複数のファイルやフォルダなどを一括して 1 つのファイルにまとめて扱うためにいくつかのフォーマットがあります。**ZIP 形式**などが用いられます。ファイルをまとめると同時にデータ圧縮なども適用されることが多いです。

2) 拡張子

ファイル名の末尾に例えば「.jpg」のようにピリオドと数文字の文字列をつけてそのファイルフォーマットを示すものを「拡張子」と呼びます。

OS は拡張子からファイルのフォーマットを判断し、関連付けられたアプリケーションソフトを呼び出します。

拡張子だけを変更してもデータそのものは変わっていないので、適切なアプリケーションソフトで開けなくなりますので注意してください。

3) ファイルの圧縮と暗号化

デジタル化された情報については、データの性質を用いて**データ量を圧縮**することが可能です。簡単な例で言えば「00000000」という 8 つの文字列は「0」が「8」個という約束で表現すれば「08」と 2 文字で表せます。同じ文字が続くことが多いテキストファイルではこのような方法でデータ量を削減できます。

- **可逆圧縮**：文書ファイルや数値データ、あるいはコンピュータのプログラムなどは圧縮したデータからもとのデータを正確に復元できる必要があります。正確に復元できるデータ圧縮を「可逆圧縮」と呼びます。
- **非可逆圧縮**：画像や音声、動画では、人の感覚の特性を利用して再生する時の品質の低下を許容することで、データ量を大幅に削減することも可能です。このようなデータの圧縮は「非可逆圧縮」と呼ばれます。画像や動画などを扱うときには、望む品質とデータ量のバランスを意識して圧縮方式のレベルを選ぶことでデータを効果的に取り扱うことができます。

また、データを保護するために**暗号化**を施す場合があります。これによりファイルそのものが漏えいしてもその内容を閲覧することを難しくすることができます。

Word や Excel などではファイルメニューで「ファイルにパスワードを設定してデータを保護する」という設定を選ぶことでデータを暗号化できます。暗号化したデータを扱うときには、容易に推察できないパスワードを使うこと、他人にファイルを渡すときは、パスワードとファイルを独立に渡す、などパスワードを適切に扱うことが重要です。

4) メタデータ

JPEG などの画像データには画像そのものに加えて、さまざまな属性情

報のデータが付加されていることがあります。このようなデータに付加する、それ自身を説明するデータのことを「メタデータ」と呼びます。デジタルカメラで撮影された写真には **EXIF (Exchangeable image file format)** という形式で撮影時刻などが記録されるメタデータが付加されます。このようなメタデータが付加されていることを理解しておかなければ、意図せずに情報を漏えいさせてしまうことがあります。

2.5.4 文字コード

ファイルを含めてコンピュータの内部では文字に2進数のコードを割り当てた文字コードを使って文字を表現し記録しています。³

1) 英数字の文字コード

英数字（ラテン文字と呼ばれます）については **ASCII (American Standard Code for Information Interchange)** コードが標準的に使われています。これは1文字に7ビット（可能な組み合わせは128）を使うことで、数字、アルファベットの大文字、小文字、記号と制御用の符号を収容したものです。日本ではこれに準拠する JIS 規格のコード(JISX0201)が使われていますが ASCII コードの「バックスラッシュ」と「チルダ (~)」はそれぞれ「¥」と「オーバーライン」に入れ替えられています。ただし、後者は Windows ではチルダが割り当てられています。

2) 日本語の文字コード

大量の漢字を扱う日本語では技術の進展とともに複数の文字コードが整備されたために、日本語を扱う文字コードの事情は複雑です。特にオペレーティングシステムによって採用されている文字コードが異なったりするために文字コードを間違えて処理して読めなくなるいわゆる「文字化け」の問題が生じています。

³ 文字コードを標準化することで情報をやりとりが可能になりますが、漢字の扱いなど、技術の進歩とともに文字コードの必要性が高まったり、業界標準と公的な機関の標準が並存したりするなどその実態は複雑です。ここでは、パーソナルコンピュータを利用する上で出会う文字コードについて要点だけを紹介します。

- **JIS X 0201**：最も古く規格化され、ASCII コードに加えカタカナなどを 7bit のコードとして定めたものです。いわゆる「半角カナ」に用いられているコードです。
- **JIS X 0208**：JIS X 0201 に 2 バイトを利用して漢字を割り当てる拡張を行った文字コードです。
- **ISO2022-JP**：電子メールなどインターネットで日本語の通信をおこなうために策定されたもので JISX0201 のラテン文字（英数字）や JISX0208 の漢字などを含んでいますが、JISX0201 のカタカナは使えません。メールソフトなどで「JIS コード」と呼ばれています。7bit での利用を想定して、文字集合を切り替える符号を途中で挿入する形で文が記述されます。
- **Shift-JIS コード**：パーソナルコンピュータで漢字を扱うために作成された文字コードです。1 バイト（8bit）で JISX0201 のラテン文字とカタカナを表記する利用と共存しつつ、文字集合を切り替える符号を挿入することなく、2 バイトの漢字を扱えるようにしたコードです。
- **EUC-JP コード**：UNIX と呼ばれるオペレーティングシステム上で広く使われた文字コードで ASCII コードと JISX0208 の漢字コードを文字集合の切り替え符号の挿入なしに使えるようにしたものです。JISX0201 のカタカナの扱いや漢字に割り当てる符号の領域が Shift-JIS コードとは異なっています。
- **Unicode**：さまざまな国の文字を統一的に扱う目的で設計された文字コードです。漢字については中国語、日本語、韓国語についてひとまとめにして扱っています。Unicode の文字を実際にコンピュータで扱うデータ列にする方法も定めており、文字を 1～4 バイトの可変長で扱う UTF-8 と呼ばれる方式が ASCII コードに上位互換であることなどから広く使われています。

3) 同じ文字の扱い

文字コードの規格は徐々に拡張されてきたため、同じ文字を異なるコードで扱うことができるようになっており、ワードプロセッサなどで文章を書くときなどは留意が必要です。以下のような運用に心がけましょう。

- 英数字は ASCII コード（半角文字）で表記し、漢字コード中の英数字（全角文字）は避けましょう。ワードプロセッサでは英語の表現に適したフォントを選べるだけでなく、綴りや文法のチェックなども ASCII コードで表記された部分に適用されます。
- JISX0201 のカナ文字（いわゆる半角カナ）は避けましょう。ISO2022-JP など文字コードによっては使えない場合があります。

4) 文字化け

Windows, macOS, Linux などオペレーティングシステムによって日本語を扱うための文字コードが異なっているため、これらの OS の間でファイルを交換したり、Web サーバと日本語のファイルをやりとりしたりする場合に文字コードの混乱（文字化け）が生じます。ファイルの内容だけでなく、ファイル名でも生じることがありますので注意が必要です。

文字化けは以下のような場合に生じます。

- 異なる OS 間でファイルをやり取りする場合
- 電子メールのメッセージ
- 文字コードを明確に指定していない Web ページの閲覧

ソフトウェアによってはデータの中で現れる 2 進数表現が文字コードによって異なることを手掛かりに、使われている文字コードを自動判定してくれるものがありますが、完全ではありません。

2.5.5 記憶装置とファイルシステム

具体的にファイルを保存する記憶装置にはさまざまなものがあります。また、ファイルを「フォルダ」などで管理し、記憶装置上に配置するための仕組みはファイルシステムと呼ばれています。コンピュータを利用するに際してはファイルがどこに保管されているのか、その記憶装置やファイルシステムの得失は何かなどを知っておく必要があります。

1) 内蔵の HDD と SSD

パーソナルコンピュータの OS やユーザのファイルなどを主に格納するための記憶装置として内蔵の HDD (ハードディスクドライブ, 磁気 of 円盤に記憶する装置)や、半導体のフラッシュメモリを HDD と同様に使

えるようにした SSD (ソリッドステートドライブ)があります。後者は前者に比べて機械的な可動部がないため動作速度が速い、衝撃に強いなどの長所がありますが、容量当たりの単価が高くなります。

2) 取り外し可能な記憶メディア

HDD や SSD を USB 経由でパーソナルコンピュータに接続する外付けのドライブに加え、フラッシュメモリに記憶する USB メモリや SD カードなどがあります。また、CD や DVD などのメディアも取り外し可能な記憶メディアとして使われます。CD や DVD については読み出し専用のもの、1 度だけ書き込み可能なもの、複数回書き込み可能なものなどさまざまな種類があります。

3) ファイルサーバ

ネットワークに接続して、ネットワーク経由でパーソナルコンピュータにファイルを提供するサーバです。NAS (Network Attached Storage)はこのようなファイルサーバの一種です。

4) ストレージサービス

OneDrive, Google Drive, Dropbox などインターネット経由で利用できるさまざまなストレージのサービスがあります。

5) ファイルシステム

HDD などの記憶装置上にコンピュータ上で大量のファイルを適切に管理するためにハードディスクなどの上にどのようにファイルの情報を記録するのが重要になります。このための仕組みを「ファイルシステム」と呼びます。例えば Windows では NTFS というファイルシステムが広く用いられていますが、他の OS では例えば MacOS の APFS (Apple File System) や Linux の Ext4(Fourth Extended File System) のようにファイルシステムとして異なる技術が採用されています。また、取り外し可能な記録メディアでは FAT32 と呼ばれるファイルシステムが利用できる機種が多いため採用されています。ファイルシステムでは単にファイルを扱うだけでなく、アクセス制限をしたり、ファイルを暗号化したり、データ

圧縮したり、あるいは記憶装置の障害に対応したりするなどの機能を有しています。

ファイルシステム上でファイル管理を行う仕組みとして「フォルダ」があります。これは「ディレクトリ」とも呼ばれ、いくつかのファイルをフォルダとしてまとめて扱うことができます。また、フォルダの中にさらにフォルダが置けるように作られており、階層的にファイルを管理できます。

2.5.6 ファイル操作

1) ファイル操作と留意点

コンピュータを用いた知的生産ではさまざまな形でファイルを利用します。Windows では Explorer を用いてファイルを操作します。

ファイルについて、どのような操作をしたいのでしょうか。以下のような操作があり、併せて留意点を述べます。

- ファイルを作る・得る：Word の文章や Excel での表計算データなどの形でファイルを作成することが多いと思いますが、ネットワークからのダウンロードすることも少なくありません。後者については
 - 著作権法に反していないなどダウンロードが適正であることや
 - ウイルスに感染していないなど安全性を確認することなども必要です。
- ファイル名を変える：先に述べた拡張子を変えてしまうとファイルを開くアプリケーションソフトとの関係づけが失われてしまいます。
- ファイルを複製する、ファイルを移動する：ウィンドウ環境でのファイルの複製と移動は似たような操作で行うことが多いので、どちらの操作をしたのかを確認する必要があります。
- バックアップを作る：別途、項目を立てて説明します。
- 圧縮・解凍する：ZIP 形式などのアーカイブファイルを作成したり、アーカイブファイルを展開したりします。
- ファイルを削除する：別途、項目を立てて説明します。
- ファイルのアクセスを制限する：重要なファイルの内容を誤って変えてしまったりすることの予防のためにアクセスを制限したい場合があります。

2) バックアップ

パーソナルコンピュータなどの利用では、文書などのファイルを誤って消してしまったり、パーソナルコンピュータやディスクが故障したりしてファイルを読めなくなったりするリスクがあります。例えば次のような場合が考えられます。

- 操作者自身が誤ってファイルを消してしまったり、別のファイルで上書きしたりしてしまう。
- パーソナルコンピュータやディスクが故障してファイルを読みだせなくなる。
- ウイルスなどの悪意のあるソフトウェアによりファイルを改ざんされる。最近では、暗号化してデータを人質にするランサムウェアなどによる攻撃もあります。

このような問題に対しては、ファイルの写しを定期的に作成する「バックアップ」を行うことが必要です。

- 何をバックアップするのか？
必要なデータを守るならデータのバックアップが、パーソナルコンピュータ自身を復元するならシステム（ディスク）全体のバックアップが必要です。
- どこに写しを作るのか？
同じディスク上にバックアップをすると復元などには便利ですがディスクが故障するとバックアップも失われます。外付けの HDD やネットワーク上のストレージにバックアップすればこの問題は避けられます。ランサムウェアではそのパーソナルコンピュータからアクセス可能なドライブを探して攻撃するものがあります。不要なときにはバックアップ用のディスクなどは接続しないことが求められます。
- バックアップを適正に管理する
バックアップを他のメディアなどに行った場合は、バックアップ自身を適正に管理する必要があります。バックアップが記録されていることを忘れて、うっかり他人に渡したり、廃棄したりすることが情報漏えいにつながってしまいます。

3) ファイル削除における注意

Windows ではファイルを削除しても復元可能なように一旦、「ゴミ箱」に移されます。ゴミ箱を空にすると、ファイルは復元できず、使用されていたディスク領域は再利用されるようになります。

ただし、書き込んだデータをすべて消去する訳ではなく、再利用できる領域として供されるだけです。データそのものは残っている可能性があります。この点は、例えばパーソナルコンピュータを廃棄した際に情報が漏えいする危険があることを意味します。廃棄に際しては、データを完全に消去する、ディスクそのものは物理的に破壊するなどの措置が求められます。

2.5.7 ファイルの共有や転送

ノート PC とスマートフォンの両方を使っている方も多いと思います。これらの間でファイルを転送したり、共有したりする方法は以下のようにさまざまに考えられます。それぞれ得失があります。

1. SD カードなど取り外し可能な記録メディアでファイルを移す。
2. パーソナルコンピュータとスマートフォンを USB ケーブルで接続してファイルを読み書きする。
3. パーソナルコンピュータのフォルダを共有設定し、WiFi 経由でスマートフォンからアクセスする。
4. LAN 上に共通にアクセスできるファイルサーバ(NAS)を設ける。
5. クラウド上のストレージサービスを利用する。

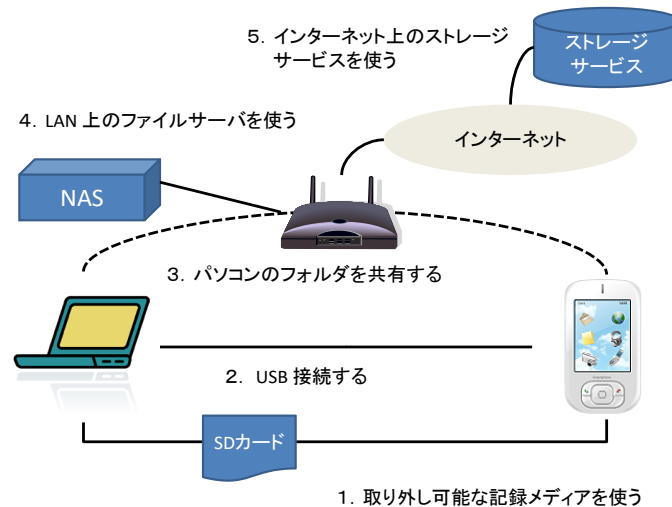


図 12 ファイルの共有と転送

2.6 OS とユーザインターフェイス

OS は利用者に以下のように統一的な操作環境を提供しています⁴。

- ウィンドウの操作
- アプリケーションソフトの起動、停止
- メニュー選択
- キーボードの操作
- 印刷
- ファイルの扱い
- コピー & ペースト

ユーザはこれらの操作をアプリケーションソフトごとに覚える必要はなく、統一的な操作でさまざまなアプリケーションソフトを利用できます。

Great Idea ! Window, Icon, Menu & Pointer

WIMP (Window, Icon, Menu & Pointer)はパーソナルコンピュータの標準的なユーザインターフェイスです。1973 年ゼロックスパロアルト研

⁴ Linux での標準的なウィンドウシステムである X-Window System は OS とは独立のソフトウェアとして構成されています。

究所 (Xerox Parc) で開発された Xerox Alto で採用されました (図 3 参照)。その後, Apple 社の Macintosh に採用され広く使われるようになってパーソナルコンピュータの標準的なユーザインターフェイスとなっています。

2.6.1 OS の利用に習熟する

OS はパーソナルコンピュータを利用する上での基本的な操作を统一的に提供しています。以下のような操作に習熟しましょう。

- ウィンドウの操作
- プログラムの起動停止
- ファイル操作
- 文字入力
- コピー & ペースト
- プリンタのプロパティでの設定
- アプリケーションソフトの実行状況の把握や動かなくなってしまったアプリケーションソフトの強制終了⁵

2.7 自立したパソコン利用者に求められるスキル

皆さんはパーソナルコンピュータ (パソコン, PC) を何のために使っていますか? ワードプロセッサ, 表計算, プレゼンテーション, 写真の整理, ビデオの閲覧, Web の閲覧, メールの読み書きなどさまざまな用途でお使いになっておられるかと思います。これらは基本的には「情報」の「加工」, 「蓄積」, 「伝達」に使います。

演習 2. パーソナルコンピュータの利用に関して以下の問に答えられますか?

- パーソナルコンピュータを新しい用途に使うためにはどうしますか?
 - アプリケーションソフトウェアの導入し, 利用方法を習得するで

⁵ Windows では「タスクマネージャ」と呼ばれるプログラムで行います。

しょう。ソフトウェアを変えることによって多様な用途に使えることはコンピュータの最も重要な特質ですが、具体的に何をすればいいのでしょうか。

- パーソナルコンピュータで作成した文書などデータはどこにあるでしょう？
 - パーソナルコンピュータ内の HDD や SSD などの記憶装置でしょうか、USB メモリや SD カードなどの取り外しのできる記録メディアでしょうか、あるいはネットワーク上のサーバでしょうか。
 - どこに保存されているのかをどうやって知ることができますか？
 - 保存場所による得失は何でしょうか。
- パーソナルコンピュータでは Windows や macOS などのオペレーティングシステム(OS)が稼働していますが OS は何をしてきているのでしょうか？
 - ユーザとして OS についてどのようなことを理解しなければならないでしょうか。

大学生になって個人でパーソナルコンピュータを専有するようになった方も多いと思いますし、一人暮らしを始めた方も少なくないと思います。このことは**パーソナルコンピュータのユーザとして自立することが求められている**ことを意味します。では自立したパーソナルコンピュータユーザに求められる知識、スキルとはどのようなもののでしょうか。単にアプリケーションソフトを使えるということだけでなく以下のようなことが求められます。

- 購入、セットアップ、廃棄
 - 予算などの制約と利用の目的に応じてパーソナルコンピュータを選定し、購入できますか。
 - 購入したパーソナルコンピュータをセットアップすることができますか。
 - 不要になったパーソナルコンピュータを適正に廃棄できますか。
- ネットワークへの接続
 - 所有しているパーソナルコンピュータを自宅や大学などで利用できるネットワークに接続できますか。

- 機器の接続
 - プリンタや DVD ドライブなどの機器を接続して利用できますか。
- アプリケーションソフトの導入，更新，削除
 - 必要なアプリケーションソフトを選定し，適正に入手し，導入したり，更新したり，削除したりできますか。
- セキュリティの確保
 - パーソナルコンピュータを安全に運用するために必要な対策を講じていますか。
 - 利用上の注意事項を知るとともに実施していますか。
 - 問題が生じた場合に適切な対応を取ることができますか。
 - 最新の動向を把握するための情報収集に努めていますか。
- データの管理
 - パーソナルコンピュータ上で扱うさまざまなデータについて，効果的，効率的に利用するためにどのような方法を取ればいいでしょうか。
 - データを保存するディスクドライブやネットワーク上のサービスを適正に選択できますか。
 - パーソナルコンピュータやディスクの故障に備えてデータの複製（バックアップ）を行っていますか。
 - 重要なデータを適正に保護する手段を知っていますか。
 - 著作権法などの法令を理解し，問題のあるデータを扱わないようにしていますか。
- ユーザの管理
 - パーソナルコンピュータを家族や友人と共用する際に，適正なユーザの権限の設定を行っていますか。
- OS やアプリケーションソフトの利用
 - OS の機能を理解し，効果的，効率的に使用できますか。
 - アプリケーションソフトやネットワークサービスについて効果的，効率的に使用できますか。

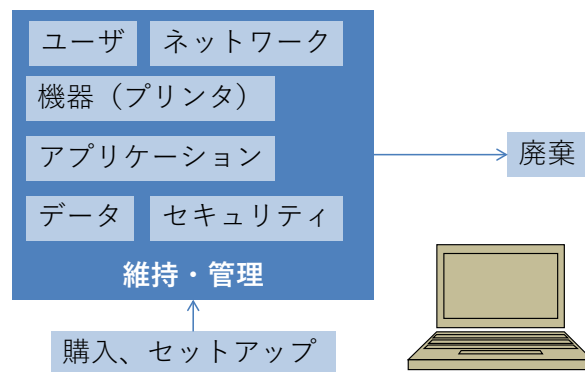


図 13 自立したパーソナルコンピュータユーザの求められるスキル

このような能力を一度に身に着けることは難しいかも知れません。パーソナルコンピュータの構成（ハードウェア，ソフトウェア）やネットワークについての基礎知識を持つことにより系統的に理解できパーソナルコンピュータの運用能力をより効率よく身に着けることができます。すなわちコンピュータを上手く使うために以下を目指しましょう。

- コンピュータとネットワークの仕組みの基本を知る
- 基本的な操作を再確認し，効率的で効果的な操作方法を学ぶ
- パーソナルコンピュータの用途として大学での学びで要求される利用方法を知る
- 何ができるようになればいいか，自身の理解やスキルを継続的に見直し，より効率的で効果的な利用方法を主体的に学ぶ

2.7.1 コンピュータを選ぶ

パーソナルコンピュータを選ぶときにはどのようなことを考えればいいでしょうか。

- CPU の性能は何に影響しますか？
- メモリの容量は何に影響しますか？
- HDD や SSD など内蔵ストレージの容量は何に影響しますか？
- ネットワークや電源への要求は？
- ディスプレイの性能への要求は？

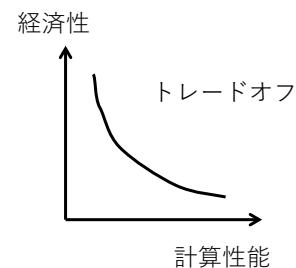


図 14 トレードオフ

- ノート PC の駆動時間への要求は？

あなたがしたいこととの関連はどうでしょうか。

- 使うアプリケーションソフト，データ量は？
- 使う場所，機会は？
- 他の機器との連携は？
- 使い方をどうやって学ぶ？
- データ移行は？

ものごとを決定する際にはトレードオフ（あちらを立てれば，こちらが立たずという関係）があります。たとえば費用と性能，可搬性のよさと使う時の操作性の良さなどです。これらを勘案して適切な決定を行うことが求められます。

2.8 キーボードのタイピング

パーソナルコンピュータを扱う上でキーボードのタイピングは人からコンピュータへの情報伝達の鍵となるプロセスです。スマートフォンの普及により、キーボードよりもタッチパネルの操作の方が多くなっていますが、大量の情報を扱う場合にはキーボードの操作は重要です。タイピングについての理解と、実際に訓練してのスキル向上はパーソナルコンピュータを使う作業の生産性を大きく向上させます。ここではタッチタイピング、英文のタイピング、キーボードを用いた Windows の操作などを紹介します。

2.8.1 タッチタイピング

タッチタイピングとはキーボードを見ることなくタイピングできることです。視線を移動することなくスクリーン上の入力結果だけを見ながら入力できるためタイピング作業を効率的に行えます。



図 15 視線移動の少ないタッチタイピング

タッチタイピングの基礎は標準的な指の使い方です。キーボードの縦の列ごとにそれを打つ指を図 16 のように決めておきます。例えば左端の「1」, 「q」, 「a」 「z」 は左手の小指で打ちます。使わない指は2列目に置くようにします。これをホームポジションと呼びます。人差し指のホームポジションが分かやすくなるようにキー「F」と「J」には小さな突起が付けられています。

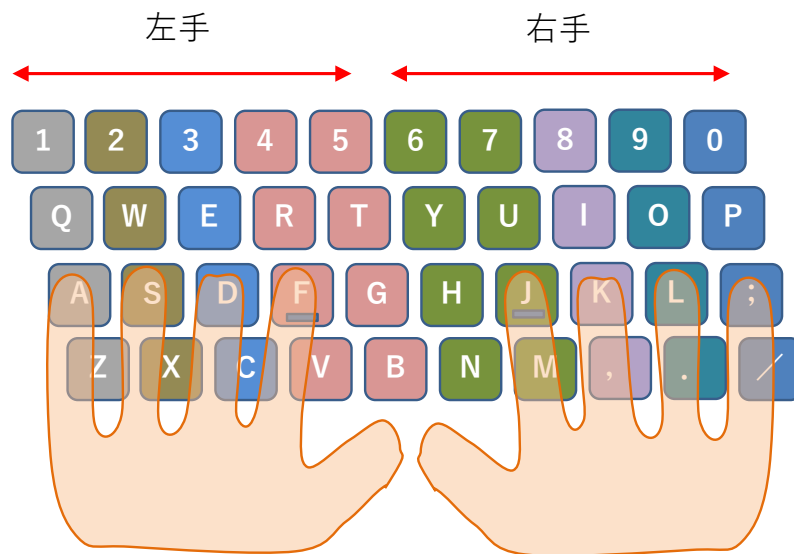


図 16 タッチタイピングのホームポジション

もちろん、タッチタイピングは練習しなければできるようにはなりません。タイピングの練習のための種々のソフトウェアがあります。無料で利用できるものや Web で練習するサイトもあるので活用して練習しましょう。

なお、現在のキーボードは下から 3 段目の文字並びから QWERTY 配列と呼ばれたりしています。タイプライタの開発の歴史の中でいろいろな並びが試され現在の配列が主流を占めるようになりました。タイプライタの歴史については参考文献[2]を参照ください。

2.8.2 英文タイピングの約束

大学では英文をタイピングすることも多くなります。Word などのワードプロセッサソフトは英文がいくつかのルールに従ってタイピングされることを前提に文書を整形したり、文法のチェックをしたりします。基本的なルールを身に付けておきましょう。

- **英文は半角文字**でタイピングしてください。
- 英文の**大文字**は図 17 のように **Shift キー**を押しながら**文字キー**を打つことで入力できます。
- Word で文章を書く場合は、**パラグラフ（段落）を変える時のみ Enter キーで改行**してください。パラグラフ中は行の右端で改行

するために Enter キーを押さないでください。

- 文を区切るためのピリオド、カンマなどの区切り記号の後には空白を1文字入れてから次の文を始めてください。ピリオド、カンマの手前に空白は入れません。
- 数字の小数点や3桁区切りのカンマは両側に空白を入れません。
- ハイフン (-) の前後には空白を入れません。⁶
- 引用符 ‘ ’ や括弧 () で文を囲うときは囲われた内側には記号と文字の間に空白を入れません。記号と外側の文字の間には空白を入れてください。



図 17 大文字は Shift キーを押しながら

2.8.3 CapsLock と NumLock

英文で大文字ばかりを打ちたいときがあります。このために使うキーが CapsLock キーです。Shift キーを押しながら CapsLock キーを押すと以後、アルファベットキーを押すだけで大文字が入力できるようになります。知らずにうっかりこの操作をすると突然、大文字ばかりが入力されるようになり困ってしまいます。もともとにもどすには同じ操作、Shift キーを押しながら CapsLock を押します⁷。

数値データの入力などで数字ばかりを続けて入れたいときに使うのが NumLock です。やはり間違えて押してしまうと困ってしまいますので、動作を確認しておきましょう。

⁶ ハイフン以外に横棒型の文字がいくつかあり、長さが異なります。Word ではハイフンの前後に空白を入れると自動的に少し長いダッシュ (–) になります。

⁷ キーを押すたびに状態が交互に代わる動作をトグル(toggle)と言います。Word などでは Ins キーは文字を挿入するモードと上書きするモードの切り替えに使います。



図 18 Caps Lock, NumLock に注意，ノート PC では NumLock は別の所に設けられています

2.8.4 Windows のキーボードでの操作

Windows でマウスを用いて行っている操作のいくつかはキーボードでも行えます。キーボードから手を離すことなく操作できるようになるので作業効率が高くなります。代表的なものとして以下の表に挙げた操作をまずは身に着けましょう。以下 Ctrl-～ とは図 19 のように Ctrl キーを押しながら～というキーを押すという操作です。

表 2 Windows のキー操作

キー操作	動作	キー操作	動作
Ctrl-C	コピー	Ctrl-Z	もとに戻す(Undo)
Ctrl-X	カット	Ctrl-Y	やり直す(Redo)
Ctrl-V	ペースト		
Ctrl + Alt + Delete	Ctrl キーと Alt キーを押しながら Delete キーを押すことでタスクマネージャの呼び出しなどができます。		



図 19 Ctrl キーの利用, Tab キーの利用

2.8.5 その場議事録

速いタイピングが役に立つ場面として会議での議事録作成があります。**会議の場で会議の内容をその場でタイプし、会議をしながら議事録を作成して行くのです。**その際、PCの画面をプロジェクタで投影し、議事録の内容を確認しながら会議を進めます。会議の終了時には、何が決定され、どの議論が積み残され、誰がどういう用務を担当するのかなどをPCの議事録を見ながら確認していきます。このようなやり方はその場で議事録を作成するので「その場議事録」とでも呼ぶべき技法です。

ラーニングコモンズを使ったグループワークなどでぜひ、トライしてみてください。

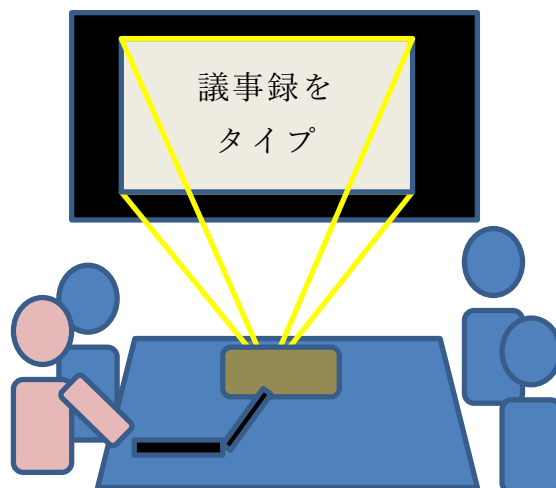


図 20 会議の場で議事録をとることで、情報共有。

タイピングは必須のスキル。

Great Idea! かな漢字変換

民族学者の梅棹忠夫が 1969 年に書いた「知的生産の技術」[1]は今でも読まれている名著ですが、その中で日本語をローマ字書きすることや、カナタイプライタを利用することなど、当時の技術でいかに日本語を扱うかを論じています。アルファベットと数字、若干の記号だけという文字種の少ない英文が早くからタイプライタを用いて文書を作成することでオフィスでの仕事を支援できていたことに対して、漢字かな混じり文を用いる日本語を機械で扱うことには困難がありました。

これが転機を迎えたのは 1978 年に森健一氏らによって開発された「かな漢字変換」という技術を用いた東芝の日本語ワードプロセッサ JW-10 の発表です[3]。そこでは今はもう当たり前の技術になっていますが、かなで入力し、コンピュータの力を借りて漢字に変換することが行われ、同音異義語の多い日本語に対して変換候補を使用頻度などで決めて提示するというものでした。

もし、この技術の開発が遅れていたら私たちはローマ字やカナ文字で日本語を書くようになっていたかもしれません。そして、それは私たちの文化にはかりしれない影響を与えていたかもしれないのです。現在ではさまざまな言語の文字でコンピュータを利用できます。日本語ワードプロセッサの開発がその先鞭をつけたという意味で人類への文化的貢献も大きいのではないのでしょうか。

2.9 ソフトウェアのライセンスと保守

2.9.1 ソフトウェアの保護

先に見たようにコンピュータはハードウェアとソフトウェアで構成されています。現代のソフトウェアは大きなものならプログラム（ソースコード）の行数が数億行にもものぼります。当然、このようなソフトウェアを開発し、保守するためには多くの人々がかかわり、多大なコストを要しています。

一方、ソフトウェアは単なるデータですので容易に複製できてしまいます。ソフトウェアの作成者の権利を守るために、**ソフトウェアは著作物として著作権法により保護**されています。

ソフトウェアの利用にあたっては、利用者は著作権者から**利用許諾（ライセンス）**を受ける形で契約を結びます。したがって、ソフトウェアの利用者はこの契約を遵守することが求められます。

コンピュータにソフトウェアをインストールする際には、利用許諾の要件が示され、これに同意した上でソフトウェアがインストールされることが多いはずです。利用許諾の内容として、利用者に何が許されているのか、提供者がどのような権利を持つのか、利用して生じた問題などに対して、提供者が責任を負わないこと（免責事項）などが示され、これに合意して利用を契約することになります。

2.9.2 ライセンスの形態

ソフトウェアのライセンスにはさまざまな形態があります。

1) コンピュータの運用形態によるもの

- **導入するコンピュータごとのライセンス**：皆さんがお持ちのパーソナルコンピュータにソフトウェアをインストールする場合に一般的に見られるライセンスの形態です。
- **フローティングライセンス**：大学に設置している端末のように多数のコンピュータを多くのユーザが使う場合、どの端末でもソフトウェアを使いたい、すべての端末で同時に稼働させることはない場合があります。このような場合に行われる契約として、どの端末で稼働させてもよいが、同時に実行できるコンピュータの台数を制限するライセンスです。ライセンスを管理するサーバを置いて、そのサーバから許諾を得た端末でだけ実際にソフトウェアが起動します。
- **サイトライセンス**：大学や学部などがその組織内で何台のコンピュータにインストールしてもよいという形態のライセンスです。

2) コンピュータの利用目的によるもの

- **学術機関向けのライセンス**：大学など、教育や研究といった公共性のある活動での利用に配慮して商業的な利用は行わないなど利用目的を限定して、安価でソフトウェアが提供されるライセンスです。

3) 無償で利用できるもの

- **フリーソフトのライセンス**：無償で利用できるソフトウェアも数多くありますが、これも多くの場合は利用許諾を受ける形で利用することになります。利用許諾に当たっては利用条件が課せられていることがあります。

2.9.3 ソフトウェアの保守の必要性

コンピュータにソフトウェアを導入した場合、いくつかの理由でこれを継続的に保守する必要があります。利用者としては提供者からの更新情報にしたがってソフトウェアの更新を行う必要があります。

- **ソフトウェアが抱える脆弱性**：ソフトウェアは外部からの攻撃に対する脆弱性の問題を抱えています。特に多くの人が頻繁に利用するソフトウェア（オペレーティングシステム、Web ブラウザ、オフィスソフトなど）は攻撃の対象ともなりやすいとされています。
- **他のソフトウェアへの依存**：アプリケーションソフトはオペレーティングシステムや共通のライブラリなどに依存して稼働します。オペレーティングシステムやライブラリが更新された場合、アプリケーションソフトが新しい状況では適正に稼働しないことも生じます。

また、すでに提供者の側で保守されていないソフトウェアを継続的に利用することは脆弱性を放置することになるので危険です。

2.9.4 ソフトウェアによる情報収集

最近のソフトウェアは提供者のサーバと連携して動かすものが多く、サーバへの連携などに際してソフトウェアの利用者の情報を提供者が収集するものも少なくありません。利用許諾の際に、どのような情報を、誰が、どのような目的のために集めるのかが示されます。ソフトウェアの導入に際してはこれらの点にも留意する必要があります。

演習 3. パーソナルコンピュータのハードウェア

あなたがパーソナルコンピュータを保有しているならご自身のコンピ

ュータについて以下の事項を調査してください。

- 搭載されている CPU
- 主メモリの容量
- 二次記憶装置（HDD や SSD）の種類や容量
- ディスプレイの解像度
- 外部に機器を接続するためのインターフェイス

演習 4. パーソナルコンピュータのソフトウェア

あなたがパーソナルコンピュータを保有しているならご自身のコンピュータについて以下の事項を調査してください。

- オペレーティングシステムの種類, バージョン
- オフィスソフトの種類, バージョン
- ウイルス対策ソフトの種類, バージョン
- 上記のソフトウェアの利用許諾条件
- 上記のソフトウェアの更新方法

演習 5. オペレーティングシステムの役割

オペレーティングシステムの役割について簡単にまとめてください。

演習 6. 外部機器の接続

あなたがパーソナルコンピュータを保有しているなら、プリンタや外部ディスプレイ、外付けの記憶装置などを接続するためのインターフェイスとしてどのようなものを備えているのか調査してください。

演習 7. プリンタの利用

あなたがパーソナルコンピュータを保有していてプリンタを接続しているなら以下の事項を調査してください。

- プリンタの接続方法
- プリンタの設定の調整方法
- インクカートリッジなど使用している消耗品とその交換方法

- 印刷を中断する方法

演習 8. 記録メディアについて

CD, DVD, USB メモリ, SD カードなどについて以下の点について調査してください。

- メディアの種類と容量, 書き込みが行えるかどうか
- メディアを介したウイルスへの感染のリスクと対策
- メディアによる情報漏洩のリスクと対策

演習 9. ファイルの管理

あなたのパーソナルコンピュータや大学の教育用 PC 端末で以下のような作業をする方法を確認してください。

- ファイルやフォルダの作成
- 拡張子を含めたファイルの情報の確認
- ファイルの移動や複製
- ファイルの削除
- 複数のファイルを一括して扱う方法
- ファイルの圧縮と解凍

演習 10. ファイルの転送・共有

以下の機器の間でのファイル転送について, 方法を検討してください。複数の方法があればその得失を考えてください。ただし, 転送元, 転送先の者以外にはそのファイルへのアクセスは許さないことを条件とします。

- 教育用コンピュータシステムの PC 端末とあなたのパーソナルコンピュータの間でのファイルなどの転送
- グループで活動する際のグループ内でのファイルの共有

演習 11. グラフィカルユーザインターフェイスの利用

あなたの使っているパーソナルコンピュータで以下の操作を行う方法を確認してください。

- ウィンドウの大きさや配置（場所や上下関係）の調整

- アプリケーションソフトウェアの起動, 停止
- マウスのボタンごとの操作
- タッチパネルの操作

演習 12. タイピング

キーボードのタイピングについて, 以下を確認してください。

- 大文字など複数のキーを用いて行う入力の確認
- タッチタイピングの指使い
- 英文タイピングのルール
- キーボードを用いたコピー&ペーストのしかた

参考文献

- [1] 梅棹 忠夫: 知的生産の技術 (岩波新書) 岩波書店 (1969)
- [2] 安岡孝一, 安岡素子著: キーボード配列 QWERTY (クワータィ) の謎, NTT 出版, 2008.
- [3] 情報処理学会, コンピュータ博物館, JW-10,
<http://museum.ipsj.or.jp/computer/word/0049.html> (2015 年 5 月 11 日アクセス)

3. ネットワークの基礎

3.1 学習の目的

インターネットで世界中のコンピュータが接続されることは大きな社会変革をもたらしています。さまざまなサービスがインターネットを通じて提供されるようになりましたが、同時に多くの問題も生じており、情報セキュリティや情報倫理について正しい知識を身に付け、それに従って行動することが求められています。本章では、このような状況を理解し、実践するために

- LAN やインターネットなどネットワークの基礎的な知識
- 情報通信技術（ICT, information and communication technologies）の利用に際してのリスクと対策

について学びます。また、現代ではさまざまなメディアを用いてコミュニケーションが行われますが、

- フォーマルなコミュニケーションで用いられる電子メールについて
- 京都大学でのサービスと学生の皆さんに求めたい事項を説明します。

3.2 仮想社会の安全運転

インターネットにより世界中のコンピュータや人々がつながった巨大な仮想社会が形成されています。コンピュータは知的能力を飛躍的に向上させる道具、言うなれば知的パワードスーツ、あるいは交通で例えるならジェット機や大型トラックといったところでしょうか。そしてインターネットは、知的パワードスーツに身を固めた人々が自在に走り回る仮想社会のハイウェイです。



したがって、様々な凄いことができるようになっていますが、仮想社会で安全運転しないと大変なことになるってしまいます。仮想社会では実社会以上に犯罪など悪意のある行為も横行していますし、多数の人々によるバッシング行為など実社会では生じにくい問題も深刻です。

このような仮想社会で活動するために情報ネットワークやそこで提供されている

サービスについて基本的な知識を持つことが求められます。

3.3 情報セキュリティ e-Learning の受講

情報倫理や情報セキュリティについては数多くの学ぶべき事項があり、この授業でも折に触れて必要な注意喚起はしますが、系統的に知識を身に着けて頂くために情報環境機構が提供する情報セキュリティ e-Learning での学習が求められています。これを完了して頂くことも本章での学習の必要事項となります。

情報セキュリティ e-Learning を修了してください

3.4 ネットワーク上の情報システム

現代の情報システムはネットワークを介してコンピュータを相互に接続することで構成されています。本書でもしばしば「サーバ」という言葉が使われていますが、これは「サーバ・クライアント型」の情報システムで「サーバ」という役割を担うコンピュータを指します。以下、情報システムの構成として、サーバ・クライアント型とピアツーピア型について紹介するとともに、しばしば使われる「クラウド・コンピューティング」を併せて説明します。

3.4.1 サーバ・クライアント型情報システム

サーバ・クライアント型情報システムでは、図 21 に示すようにサービスを中核的に提供する「サーバ(server)」と呼ばれるコンピュータにサービスを受けるコンピュータ（クライアント, client）が接続する構成をとります。パーソナルコンピュータを用いたネットワーク利用には後で述べる IP アドレスを自動付与する DHCP サーバ、ドメイン名から IP アドレスを検索する DNS サーバ、Web ページを提供する Web サーバなど、さまざまなサーバに接続してそれぞれのサービスを受けており、重層的なサーバ・クライアント型の情報システムの中で動作しています。

3.4.2 ピアツーピア型情報システム

サーバ・クライアント型の情報システムではサーバという中核的なコンピュータに他のコンピュータがクライアントとして接続する形でサービスを受けますが、この

ほか，それぞれのコンピュータが直接，相互に接続してサービスを提供しあう方式として図 22 のようなピアツーピア(peer to peer, P2P) 型のシステム構成法があります。¹

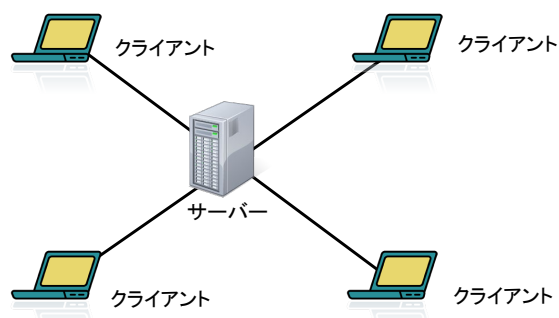


図 21 サーバ・クライアント型情報システム

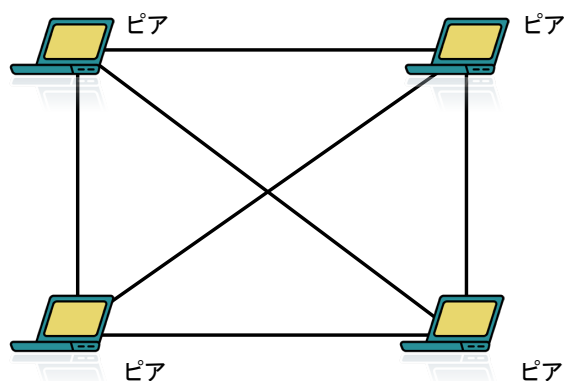


図 22 ピアツーピア型情報システム

3.4.3 クラウド・コンピューティング

大学や企業など組織の情報システムは従来，組織内にサーバを設置することで運営されてきましたが，近年ではデータセンター上で稼働しているさまざまなサービスを利用するようになってきました。図 23 に示すこのような情報システムの形態をクラウド・コンピューティング(cloud computing)といいます。クラウドという名称は図解する差にインターネットを「雲」に例えて描くことから来ています。クラウド・コンピューティング型で提供されているサービスをクラウド・サービスと呼びます。

¹ P2P 方式でファイル交換を行うソフトウェアに関しては，大量の通信が発生すること，著作権侵害を生じやすいこと，ウイルス感染の危険が高いことなどの問題が指摘されています。京都大学では KUINS III からの P2P ファイル交換ソフトの利用を禁止しています。

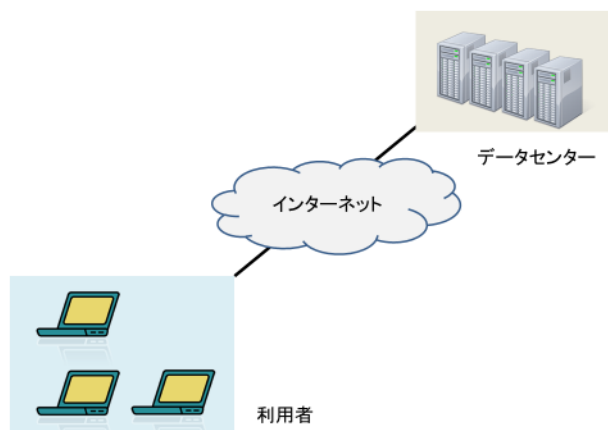


図 23 クラウド・コンピューティング

3.5 ローカルエリアネットワーク (LAN) の基礎

パーソナルコンピュータなどのネットワークへの接続は

- 有線 LAN, 無線 LAN (Wi-Fi) などへの接続と
- インターネットへの接続

の2段階になっています。以下, LAN とインターネットの基礎的事項について説明します。

3.5.1 LAN の種類

ローカルエリアネットワーク (LAN, local area network)とは建物内の狭い範囲内でコンピュータなどを相互に接続する通信ネットワークのことを言います。地域や地方などより広い範囲で構築されるネットワークを広域ネットワーク (WAN, wide area network)と呼びます。LAN には幾つかの方式がありますが,ここでは有線の LAN と無線の LAN について代表的なものを説明します。

1) 有線 LAN

有線 LAN として用いられている方式としてイーサネット (Ethernet)と呼ばれるものがあります。

- 同じ RJ-45 型コネクタを使いますが, 表 3 に示すように通信速度の異なるいくつかの規格があります。通常, 自動的にどの規格に従うか設定されます。
- 規格によって要求されるケーブルの品質 (カテ



図 24 RJ-45 型コネクタ

ゴリと記載されます) が異なっています。

- ハブやスイッチと呼ばれる機器を介してパーソナルコンピュータなどの機器を相互に接続するネットワークを構成します。

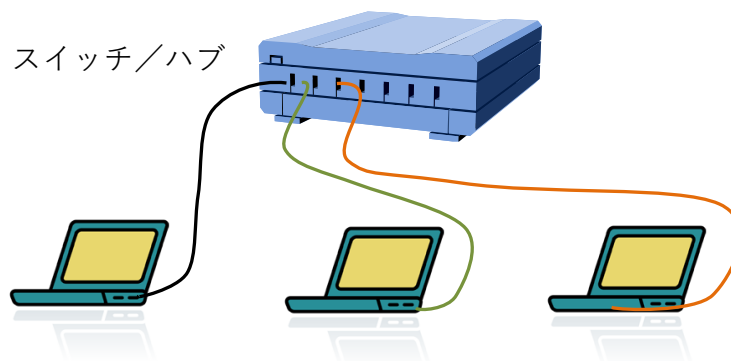


図 25 有線 LAN (イーサネット) の構成

表 3 主な有線 LAN の規格

規格	通信速度	使用するケーブル
10 BASE-T	10 Mbps	カテゴリ 3
100 BASE-TX	100 Mbps	カテゴリ 5
1000 BASE-T	1000 Mbps	カテゴリ 5 e

2) 無線 LAN (Wi-Fi)

無線 LAN は通常、無線 LAN のインターフェイスを持つパーソナルコンピュータなどの機器がアクセスポイントという機器を介して相互に無線通信する LAN です。

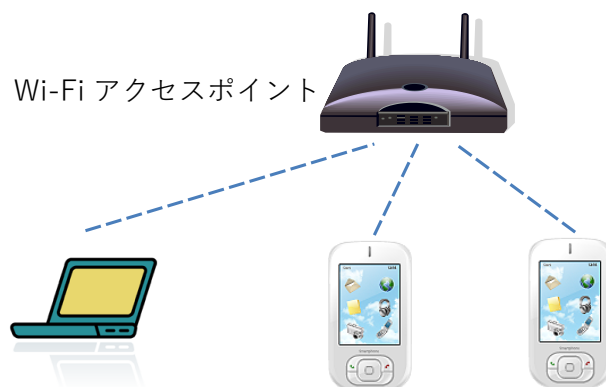


図 26 無線 LAN (Wi-Fi) の構成

無線 LAN は使用する電波の周波数帯、通信速度によっていくつかの規格があります。通常、パーソナルコンピュータとアクセスポイントの間で自動的に適用する規格が決定されます。

表 4 主な無線 LAN(Wi-Fi)の規格

規格	周波数帯	最大通信速度
802.11a	5GHz 帯	54Mbps
802.11b	2.4GHz 帯	11Mbps
802.11g	2.4GHz 帯	54Mbps
802.11n	2.4GHz 帯, 5GHz 帯	600Mbps
802.11ac	5GHz 帯	6.9Gbps

SSID, ESSID: 同じ空間で複数の無線 LAN が混在できます。SSID, ESSID は個々の無線 LAN グループを識別するために付けられている名前です。

3) 無線 LAN での暗号化

無線 LAN では電波が空間中を伝わるため容易に他者に盗聴されます。これを防ぐために暗号化が行われます。暗号化された無線 LAN には SSID に加え暗号化キー

が接続に必要です。自宅などでアクセスポイントを設置する場合は WEP など脆弱性が指摘されている暗号化方式は使わず WPA2 (AES) を使ってください。

表 5 無線 LAN の暗号化の規格

暗号化の種類	備考
WEP	初期の方式，脆弱性が指摘されている
WPA	WEP を改良，TKIP と呼ばれる暗号化方式を利用
WPA2	WPA の強化版，より強力な AES 暗号に対応

暗号化を行わない LAN を安全に利用するためには VPN (Virtual Private Network) 接続などを用います。

3.5.2 MAC アドレス

LAN の機器ごとにあらかじめメーカーによって固有の番号(MAC アドレス，Media Access Control Address)が定められ，多くの場合，予め当該の機械に設定されています。48 ビット（6 バイト）のアドレスが用いられ，表記は 1 バイトを 16 進数 2 桁で表したものを「-」や「:」でつなぐ形で

〇〇-〇〇-〇〇-〇〇-〇〇-〇〇

というような形で表されます。「物理アドレス」という呼び方で MAC アドレスを表すこともあります。

Windows ではコマンドプロンプトを起動し，getmac というコマンドを入力することで，そのコンピュータのネットワークインターフェイスの MAC アドレスを調べることができます。

3.5.3 LAN の構成と機器

- ハブ，スイッチ：有線 LAN で接続する機器間のパケット（LAN ではフレームと呼ばれる）を配送する装置です。電気的な信号を増幅するだけのものと，パケットを内部で保持して，再度送り出す構成をしているものがあり，後者をスイッチと呼ぶ場合があります。
- アクセスポイント：無線 LAN で接続するコンピュータなどの機器どうしや有線 LAN との間でフレームを配送する装置です。

- ルータ：LAN と外部の回線(WAN, Wide Area Network)や別の LAN との間に入って IP パケットを配送する装置です。
- DHCP サーバ：LAN に接続する機器に対して自動的に IP アドレスを割り当てる装置です。DHCP については後で説明します。
- WAN 接続の機器：光ファイバ回線用の「光回線終端装置 (ONU, Optical Network Unit)」やケーブルテレビ回線用の「ケーブルモデム」など回線種別に応じて外部のネットワーク(WAN)に接続するための装置です。

これらの機器を用いて LAN は図 27 の左の図のように構成されますが、Wi-Fi ルータなどでは小規模なネットワーク用にこれらの機能を含む形で同図右のように構成されています。

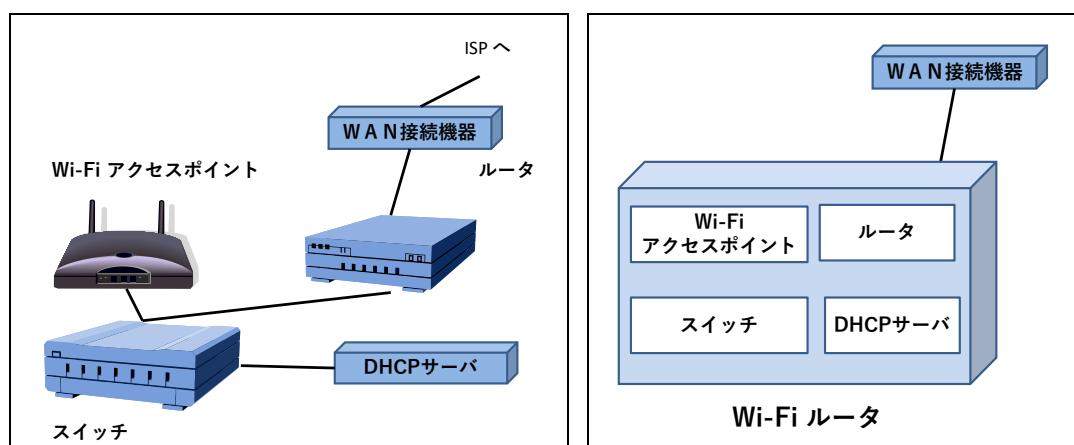


図 27 LAN の構成機器

3.5.4 テザリング

スマートフォンにはスマートフォンを介して PC などをインターネットに接続させる「テザリング」と呼ばれる機能があります。Wi-Fi を使ったテザリング¹はスマートフォンに Wi-Fi アクセスポイント、ルータ、DHCP サーバの役割をさせることで PC などを Wi-Fi 経由でインターネット接続させる仕組みです。

¹ Wi-Fi のほかにも USB や Bluetooth を用いたテザリングもあります。

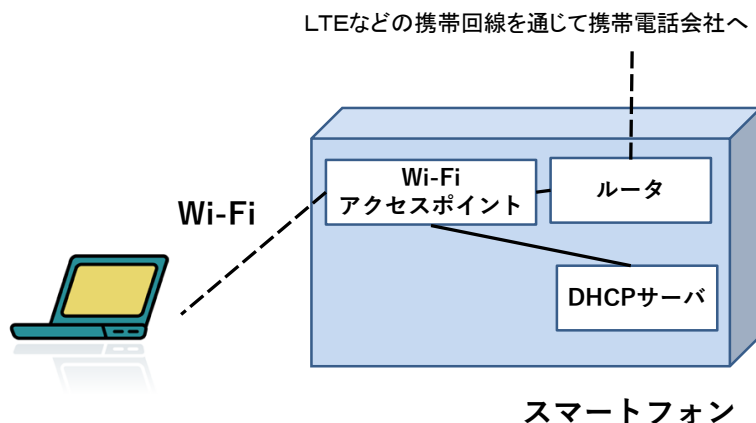


図 28 スマートフォンを用いた Wi-Fi テザリング

3.5.5 VPN (Virtual Private Network)

大学など、組織内のネットワークでは、組織内からだけアクセスを許す WWW サーバなどが運用されていることがしばしばあります。このようなサーバへは組織の外から直接、アクセスすることはできません。

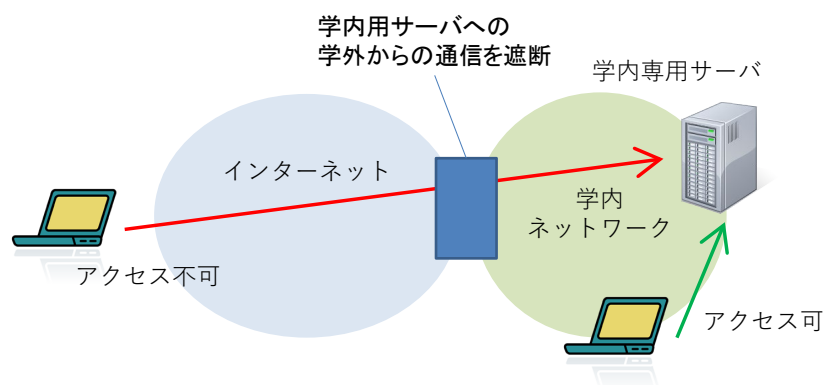
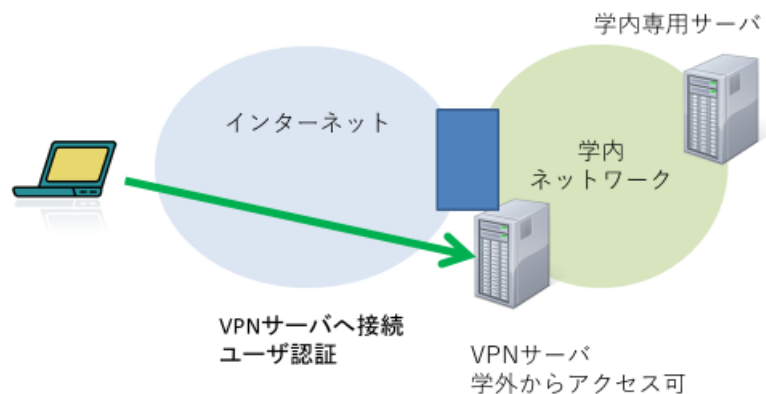


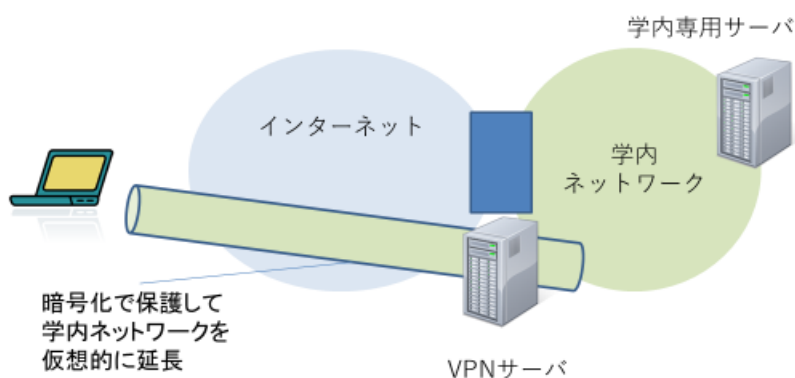
図 29 組織内にだけアクセスできるサーバ

このような組織内のネットワークにインターネット経由で安全に接続する仕組みとして VPN (Virtual Private Network) があります。パーソナルコンピュータなどでインターネットから組織内のネットワークへの VPN を用いたアクセスは以下のような手順で行われます。

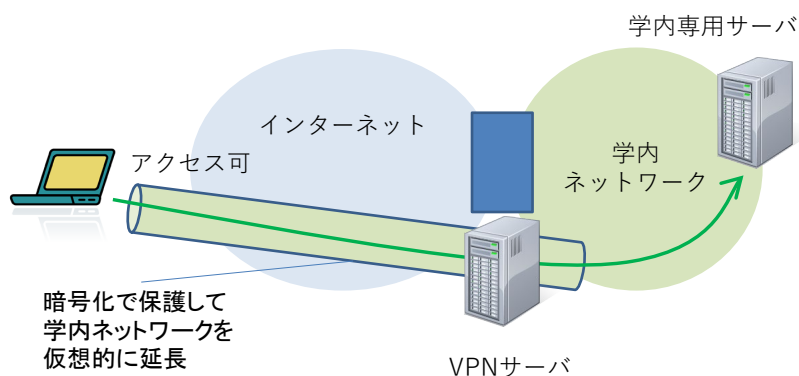
1. まず、インターネットに向けて公開された VPN サーバにアクセスし、アクセス権を持つユーザであることの認証を受けます。



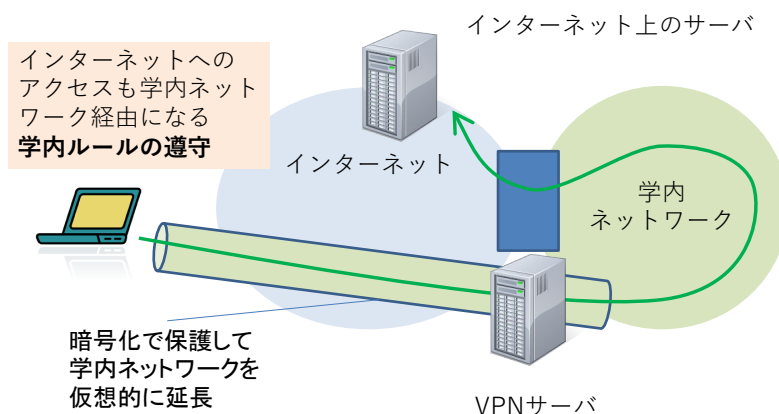
2. VPN サーバは接続してきたパーソナルコンピュータとの間に暗号化により保護した通信経路をインターネット上の仮想の経路として、パーソナルコンピュータから学内のネットワークへの通信を許可します。



3. この経路を経由してパーソナルコンピュータから学内のサーバなどへアクセスします。



注意： 組織内のネットワークに VPN 接続すると、VPN 接続元のパーソナルコンピュータからのインターネットへのアクセスも VPN 経由で行われ、その組織の制限が適用されます。その組織で禁止されている通信は行ってはいけません。



3.6 インターネットの基礎

インターネットを介して図 30 のように様々な機器が接続されています。以下ではインターネットについて基礎的な事項を学びます。

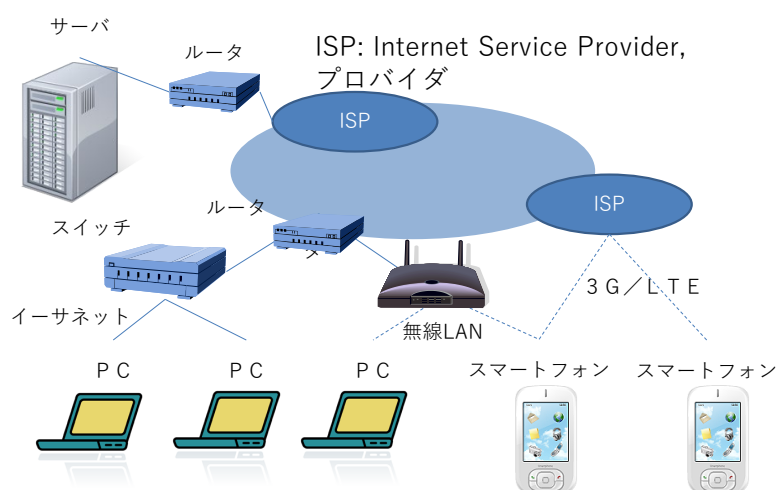


図 30 ネットワークを構成するさまざまな機器

3.6.1 インターネットの仕組み 1: TCP/IP

インターネットは LAN などさまざまな種類のネットワークを介して使える共通の通信の仕組みを提供しています。情報はパケットという単位で配送されます。これにより 1 つの回線で多くの相手と並行して通信できます (図 31 参照)。

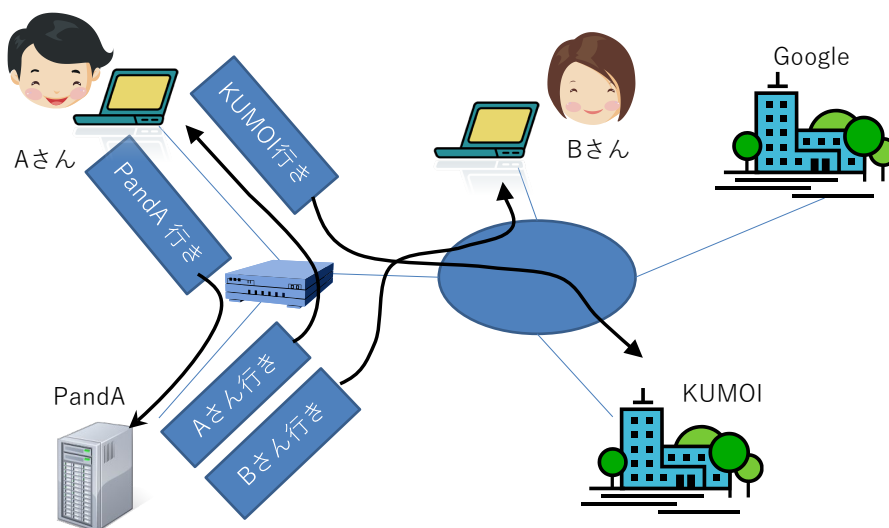


図 31 パケット通信

インターネットは末端から末端へのパケットの配送が基本です。

- IP アドレス: それぞれの末端の機器を識別するためのアドレス (番地) が「IP アドレス」です。
- IP (Internet Protocol): パケットレベルの配送だけを担当する通信規約 (プロトコル, Protocol) です。パケットを最善の努力で配送しますが, パケットの到着や順序などは保証しません。
- TCP (Transmission Control Protocol): 末端から末端へデータが順序を守って確実に届くことを実現する通信規約です。IP レベルではこのことは保証されませんので, 再送や順序の復元などを行っています。1 台のコンピュータが TCP 上で同時に複数のアプリケーションの通信を行うために IP アドレスに加え, ポート番号を用いて, 通信を識別します。
- UDP (User Datagram Protocol): IP と同じようなパケット (名称のデータグラムはパケットのことを意味します) 単位の配送をユーザ向けに提供する通信規約です。TCP と同様, IP アドレスに加え, ポート番号も用いて通信を識別します。
- HTTP (Hyper Text Transfer Protocol): TCP を用いて World Wide Web のサービ

スを行うためのプロトコルです。

このほかにも，TCP/IP を基盤にさまざまな通信規約（プロトコル）が定められています。

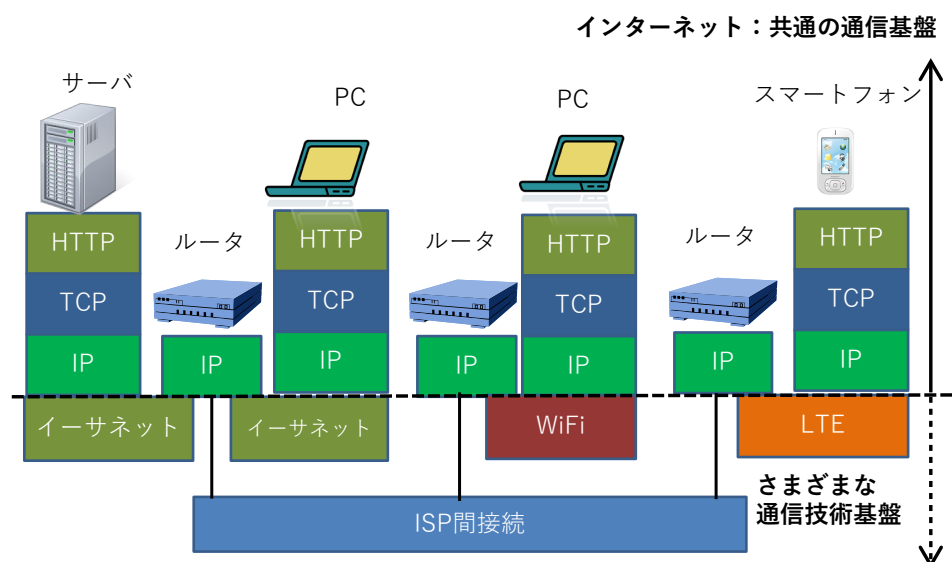


図 32 インターネットを構成する通信規約（プロトコル）

Great Idea! インターネットプロトコル

インターネットの基盤であるパケットを配送するインターネットプロトコル (IP) は

「最善の努力で配送」する。到着すること、パケットの順番を守ることとも保証しない。

と定められています。

ネットワークの途中では，このことだけを実現すればよいので，中継の仕組みが簡単になります。長いメッセージを，複数のパケットを使って，到着や順番を保証して通信することは末端での役目とされ TCP が担います。IP と TCP がインターネットを代表する通信規約であるので TCP/IP と称されます。

このような仕掛けで利用者がさまざまなサービスのためのプログラム(ソフトウェア)を開発できることがインターネットの発展を支えました。

3.6.2 LAN 内の通信とルータの役割

イーサネットや Wi-Fi など LAN で相互に接続された機器の間では送り側や受け側の MAC アドレスを含めたフレームで相互に通信します。LAN 内で IP アドレスを用いた通信は以下のように実現されます。

1. 受け手の IP アドレスを持つ機器の MAC アドレスを知る。これは LAN 内の機器全体に同報通信機能を用いて受け手の IP アドレスを持つ機器の MAC アドレスを照会することで行われます（ARP, Address Resolution Protocol という方式が使われます¹⁾）。
2. 受け手の MAC アドレスを指定した LAN のフレームのデータ部分としてインターネットプロトコルとしてのパケットを埋め込んで送出します。
3. 受け手は LAN のフレームのデータを取り出して IP のパケットとして処理を行います。

世界中の WWW サーバなど、LAN の外側にある IP アドレスを持つ機器宛の通信は受け手としてルータ（ゲートウェイ）を指定して同様の手段で送ります。

ルータは受け取った LAN のパケットから IP パケットを取り出し、WAN など転送先のネットワークの方式に合わせてパケットを再度、埋め込んで送出します。

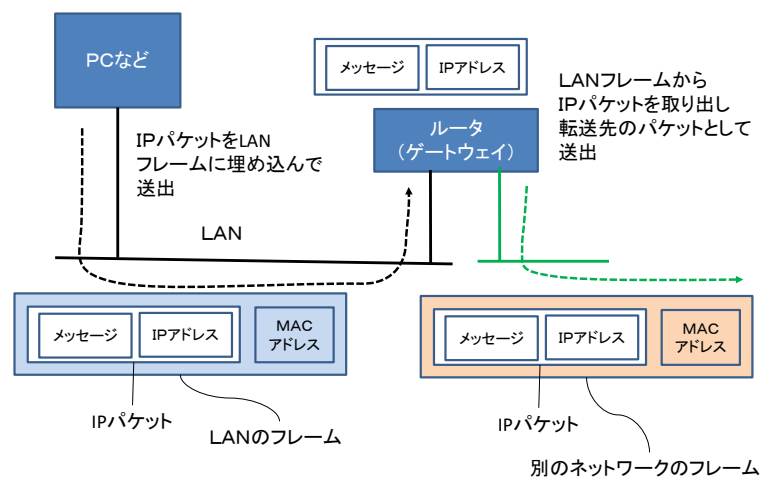


図 33 ルータの役割

3.6.3 IP アドレス

1) IP アドレスとその表記

インターネットプロトコルでは通信の相手先や送信元を示すために IP アドレスという番号を使います。我が国で広く利用されている IPv4（後述）では IP アドレ

¹ ARP は後述の IPv4 で用いられる方式で、IPv6 では Neighbor Discovery Protocol (NDP) が同様の機能を担います。

スは 4 バイトで構成され、各バイト（0～255 の値をとります）を 10 進数で表記したものをピリオド「.」でつないで表記します。例えば

192.0.2.0

などです。

2) IP アドレスの割り当て

IP アドレスは個々のコンピュータを識別し、パケットを配送するためのアドレスですからネットワーク内で（グローバル IP アドレスなら世界的に！）系統的に割り当てることが求められます。皆さん方が使うパーソナルコンピュータでは手作業で明示的に IP アドレスを割り当てることはほとんどないはずです。これはコンピュータをネットワークに接続したときに IP アドレスを自動的に割り当ててくれるサーバと通信してアドレスを取得するようにネットワークが運用されているためです。このための通信規約（プロトコル）を DHCP (Dynamic Host Configuration Protocol) と呼び、これに従って IP アドレスを割り当てるサーバを DHCP サーバと呼びます。

3) グローバル IP アドレスとプライベート IP アドレス

IP アドレスには全世界で系統的に割り当てて、インターネット上でパケットが配送されるグローバルアドレスのほかに、閉じた範囲で自由に使えるかわりに、その範囲を超えて配送してはいけないプライベート IP アドレスが定められています。プライベート IP アドレスで運用されるネットワークをプライベートネットワークと呼びます。

4) グローバル IP アドレスとプライベート IP アドレスの得失

WWW サーバなどで外部にサービスするためにはグローバル IP アドレスで運用されるネットワークに接続することが必要ですが、その分、外部からの攻撃も受けやすく、情報セキュリティ面での厳しい対策が求められます。

他方、プライベート IP アドレスで運用されるネットワークは申請による割り当てが必要なグローバル IP アドレスとは異なり、組織などで自由に利用できること、多くのアドレスを利用できることが利点です。また、プライベート IP アドレスはグローバル IP アドレスの枯渇への対応として利用されているものですが、プライベート IP アドレスで運用されている機器には外部から直接は接続できないということもあり、オフィスや家庭などでコンピュータを利用する際に用いられます。

ただし、多くのネットワークで WWW サーバへのアクセスなどのため、後述のような方法で、プライベート IP アドレスのネットワークとグローバル IP アドレスの

ネットワークを橋渡ししています。プライベート IP アドレスで運用しているコンピュータへは通常、直接、外部からアクセスはできません。しかしながら、ウイルス感染などにより悪意のあるソフトウェア（マルウェア）をコンピュータ上で稼働させることで、コンピュータ側から攻撃者のコンピュータに接続させて遠隔操作を可能にされます。このことを理解して安全なコンピュータの運用に留意が必要です。

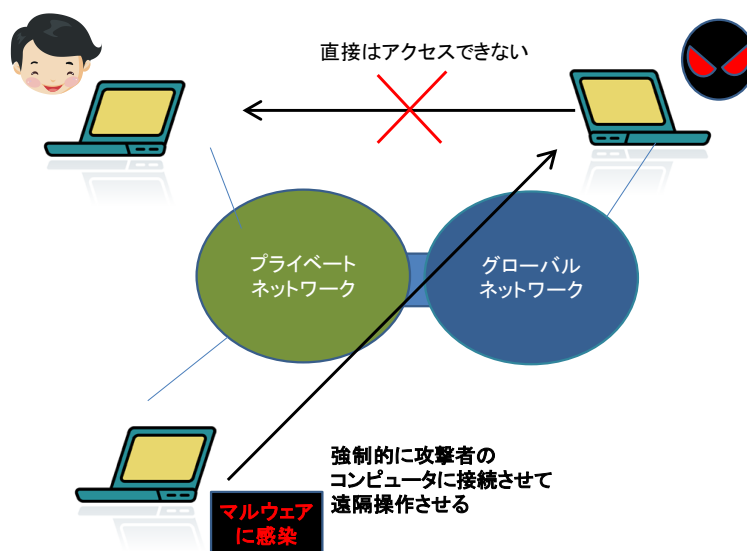


図 34 プライベートネットワーク内のコンピュータへの攻撃

一般家庭(個人)で使う有線、無線のルータでも、それに接続するパーソナルコンピュータやスマートフォンには基本的にプライベート IP アドレスが使われています。

5) プライベート IP アドレスとして使われるアドレス

プライベート IP アドレスとして使用できるアドレス（インターネット上には存在しないことが保証されているアドレス）は以下です。

- 10.0.0.0 - 10.255.255.255
- 172.16.0.0 - 172.31.255.255
- 192.168.0.0 - 192.168.255.255

指定されている範囲から上段ほど大きなプライベートネットワークで、下段ほど小さなプライベートネットワークで利用されます。

6) プライベートネットワークからインターネットへの橋渡し

プライベート IP アドレスを持つ機器はインターネット上の WWW サーバとは

直接は通信できません。インターネット上の WWW サーバへのアクセスは以下のような方法で中継することで実現します。

(a) アドレス変換による方式

ルータなど中継装置で IP パケットについて、プライベート IP アドレス（とポートの組み合わせ）に対して、グローバル IP アドレス（とポートの組み合わせ）の変換テーブルを設けてパケットを書き換えて転送する方式で、NAT (network Address Translation) (ポートを組み合わせる場合は NAPT (Network Address Port Translation)) と呼ばれます。

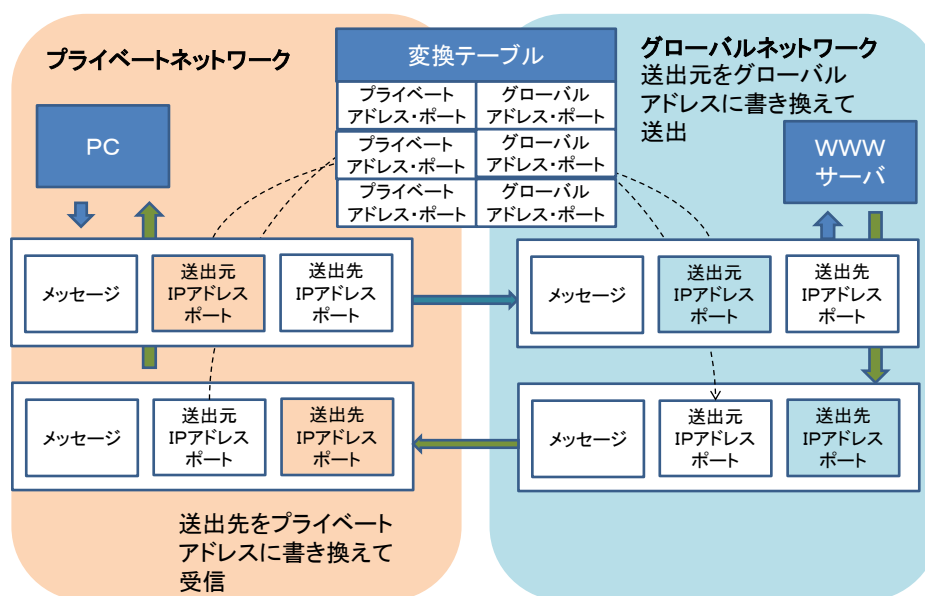


図 35 アドレス変換 (NAPT)

(b) 代理サーバによる方式

プライベートネットワークとグローバルネットワークの両方に接続したサーバを設置し、プライベートネットワークアドレス側からのグローバルネットワークへの WWW サーバのアクセスを代理サーバ(プロキシサーバ)で受け、代理サーバから実際の WWW サーバにアクセスし、その結果をプライベートネットワーク側の接続元に送り返す方式です。

3.6.4 IPv4 と IPv6

現在、広く使われているインターネットプロトコルはバージョン 4 (IPv4) と呼ばれるものです。これまでの説明はもっぱら IPv4 を想定して述べてきました。IPv4 ではアドレスを 4 バイト (32 ビット) で表現していたことから、インターネットの利用の急拡大に伴い割り当て可能なアドレスが枯渇しています。IPv4 の後継として定

められたプロトコルがインターネットプロトコルバージョン 6 (IPv6) です。IPv6 ではアドレスに 16 バイト (128 ビット) を用いて、アドレス不足の問題に対応しています。IPv6 の利用も徐々に進んでいます。

IPv6 のアドレスは 16 ビット単位ごとに 16 進法 4 桁で表記したものを「:」で区切ります。ただし、このままでは表記が長くなるため 4 桁の区切りごとの先頭の 0 は省略化とし、すべて 0 の区切りがいくつか続く場合には 1 回に限り :: と省略できるなどのルールで表記されます。

例えば

2001:0db8:0000:0000:0001:0000:0000:0001

は

2001:db8::0:1:0:0:1

と短縮して表記されます。

3.6.5 インターネットの仕組み 2:ドメインネームシステム(DNS)

IP アドレスは人には分かりにくいので、人に分かりやすい命名としてドメイン名を使います。ドメイン名から対応する IP アドレスを検索する分散的なデータベースがドメインネームシステム DNS です。ドメイン名から IP アドレスを検索してくれるサーバをネームサーバ (DNS サーバ) と呼びます。接続するネットワークによってネームサーバを自動的に取得してくれる場合と、明示的に設定しなければならない場合があります。

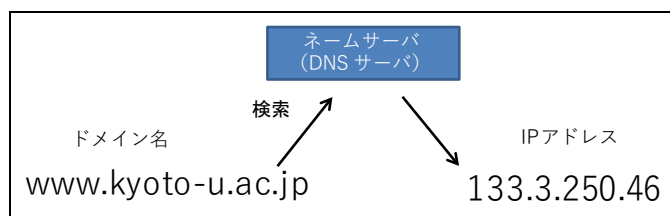


図 36 DNS によるドメイン名から IP アドレスへの変換

3.6.6 WWW の仕組み

World Wide Web は HTTP (Hyper Text Transfer Protocol) というプロトコルを用いて Web ページにアクセスする仕組みです。その特徴は

- HTML (Hyper Text Markup Language) という形式でページが記述されています。
- URL (Uniform Resource Locator) という記法でウェブサイトやその上に置かれたページの所在などを統一的に記述します。
- ハイパーリンクとしてページ内で他のページへのリンクを書くことができます。

Web サイト（サーバ）にアクセスするためのアプリケーションソフトは「Web ブラウザ」と呼ばれます。主な Web ブラウザとしては

- Microsoft 社が提供する Internet Explorer や Microsoft Edge
- Mozilla Foundation が提供する Mozilla Firefox
- Google 社が提供する Google Chrome
- Apple 社が提供する Safari

などがあります。

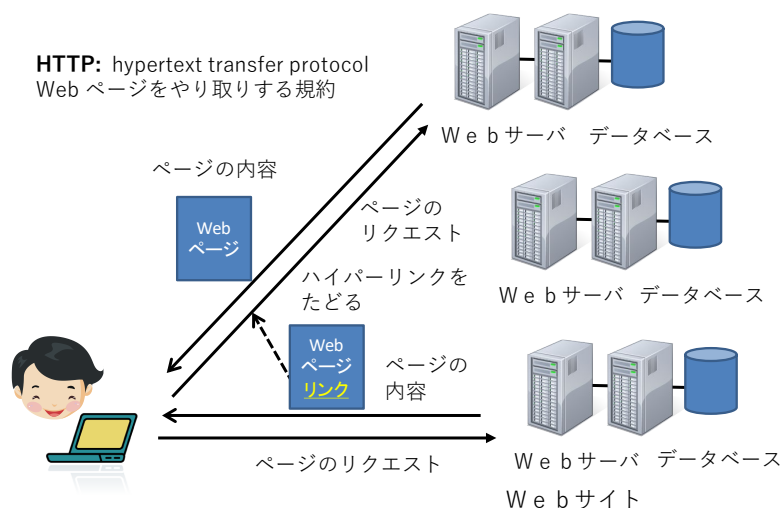


図 37 World Wide Web の仕組み（ハイパーリンク）

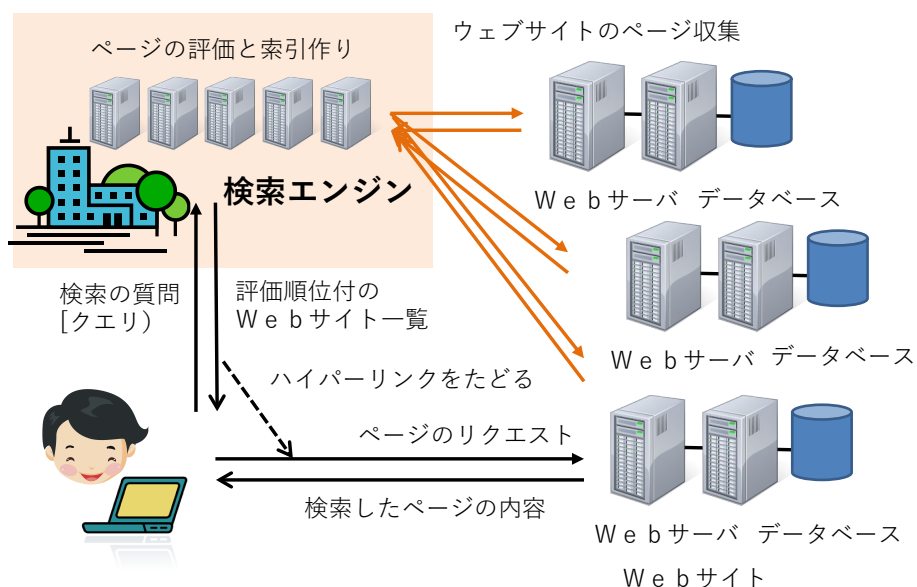


図 38 World Wide Web の仕組み（検索エンジン）

現在では World Wide Web 上で様々なサービスが提供されています。例えば

Google などの検索エンジンでは世界中の WWW サイトから情報を自動的に収集し、そのページの重要性を評価したり索引を作ったりして、検索の要求に対して該当するページの候補を提供しています。

3.6.7 電子メールの仕組み

電子メールはインターネット上の電子メールサーバを相互に接続してメッセージを配送する仕組みです。ユーザがメールを送受信する方法としては

- 電子メールサービスを提供するサイトの Web ページを介して Web ブラウザでアクセスする
- Microsoft 社の Outlook や Mozilla Foundation の Thunderbird などコンピュータでメールソフトを起動してアクセスする

という 2 種類の方法が主に使われます。

後者では

- メールを受信するためのプロトコルとして
 - POP (Post Office Protocol)
 - IMAP (Internet Message Access Protocol)
 の 2 通りがあります。
- 送信するためのプロトコルとしては SMTP (Simple Mail Transfer Protocol) が用いられます。

メールソフトを用いるには、これらについてメールサービスの指定するサーバとプロトコルを設定しなければなりません。

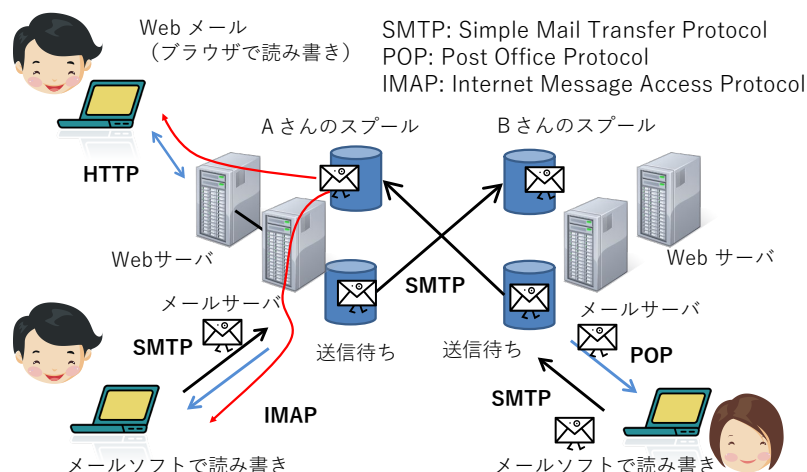


図 39 電子メールの仕組み

3.6.8 インターネットでの暗号の利用

TCP/IP によるインターネットの通信は平文で行われます。通信経路上でメッセージは特に保護されていません。内容を途中で覗き見されないためには暗号を利用します。

- インターネット (WWW) 上での暗号通信として TLS/SSL¹ が良く使われます。
- TLS/SSL での通信に対応している WWW サイトの URL は “HTTPS” から始まっています。
- POP3S, IMAP4S, SMTPS: それぞれ POP3, IMAP4, SMTP でのメールの通信を TLS/SSL を使って暗号化する方式です。POP3 や IMAP4 の 3 や 4 はそれぞれのプロトコルのバージョンを表します。TLS/SSL では最初に次に述べる公開鍵暗号を使って共有鍵暗号の鍵を交換し、通信の内容は共有鍵暗号を用いる構成を取っています。

1) 公開鍵暗号

公開鍵暗号は安全なインターネットの利用を支える極めて重要な技術です。通常の暗号は暗号化と復号化に同じ鍵（共有鍵）を用います。これに対して、公開鍵暗号とは金庫に例えると、閉める鍵と開ける鍵が異なった仕掛けで、しかも片方の鍵や暗号化されたメッセージから他方の鍵を推測することが極めて難しい（途方もない計算量を必要とする）ように作られています。

2つの鍵、A、Bを用いるとして片方を公開し、他方を秘密にします。例えば

- 鍵A：公開
- 鍵B：秘密

これを用いると

- 暗号通信：以下のような手順で誰でも秘密鍵 B を持つ者と暗号化通信ができます。
 - 公開の鍵Aで暗号化（誰でもできる）→ 鍵Bで復号化（秘密鍵の所有者のみ可能）
- 電子署名：以下のような手順で誰でも公開鍵を用いて秘密鍵 B からのメッセージを読むことができます。誰でも読めるため暗号化には使えませんが、この

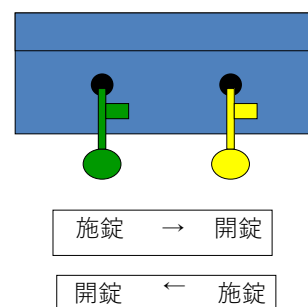


図 40 公開鍵暗号

¹ TLS は Transport Layer Security, SSL は Secure Sockets Layer の略で、歴史的には SSL から TLS へと発展しました。両方の表記が用いられているので、ここでは一括して TLS/SSL と記します。

ようなメッセージは秘密鍵 B を持っている者にしか作れないため署名の役割を果たします。

➤ 秘密鍵 B で暗号化（所有者のみ可能）→

公開鍵 A で復号化（誰でもできる）

という運用により，不特定多数の通信者を相手に暗号通信や電子署名を行うことができるのです。

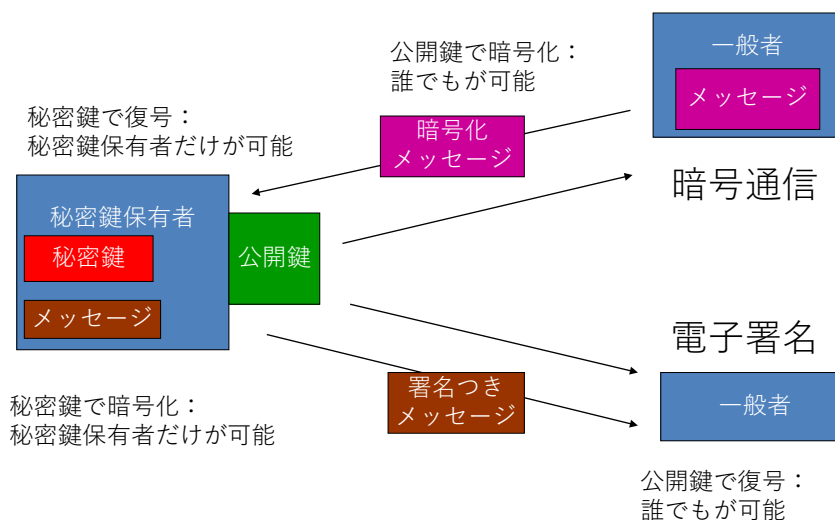


図 41 公開鍵暗号を用いた暗号通信と電子署名

Great Idea! 公開鍵暗号

公開鍵暗号が開発されるまでの暗号方式は暗号化と復号化に共通の鍵を用いていました。このため鍵を受け渡すことが難しく，不特定多数を対象にした暗号通信は困難でした。現在，広く用いられている公開鍵暗号は巨大な素数の積を因数分解することが極めて難しいという原理を用いています。

2) 認証局と証明書

公開鍵暗号を用いて暗号通信ができて，通信する相手が信用できるかどうかは別の問題です。公開鍵の発行者の正当性を知るための仕掛けが「認証局と証明書」で，印鑑証明の仕掛けを電子的に実現したものです。

- 証明書を発行する認証局を作り，その公開鍵を予め何らかの方法で利用者に配布します
- 証明を受けようと思うものは，公開鍵を認証局に登録します。これに対して，認証局は電子署名した証明書を発行します
- 証明を受けたものは公開鍵とともに証明書を利用者に示します

- 利用者は公開鍵の正当性を認証局の署名により確認します

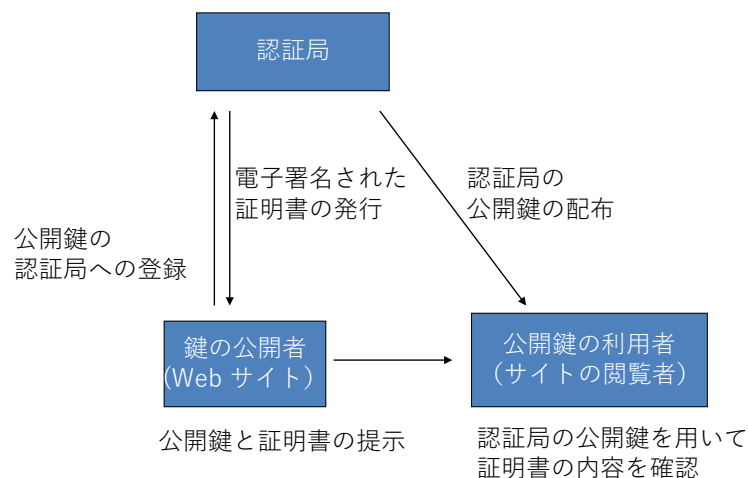


図 42 認証局と証明書

3) TLS/SSL 利用と公開鍵証明書の確認

Web サイトへのアクセスを暗号化するのに用いられている仕組みが TLS/SSL です。アクセスしているサイトへの通信が暗号化されていることと、そのサイトの公開鍵が適切に証明されていることを確認しましょう。



図 43 WWW での暗号通信と証明書の確認

3.7 パーソナルコンピュータのネットワーク接続

先にも述べたようにパーソナルコンピュータのネットワーク接続は

- LAN への接続
- インターネットへの接続

の2段階になっています。

- まず、有線 LAN、無線 LAN など LAN への接続が行われます。
- LAN 上にあるサーバから、インターネット接続のためのアドレス(IP アドレス)が割り当てられ、その他の必要な情報が提供されインターネットへの接続が行われます。

通常はこの両方は一括して行われますが、LAN 上のサーバと適切に通信ができていない場合や間違った LAN に接続した場合は LAN には接続できているが、インターネットへの接続は行えない、というような状況になります。

さらに、WWW サーバなどの URL に含まれるドメイン名（例えば `www.kyoto-u.ac.jp`）は DNS (ドメインネームシステム) を用いて IP アドレスに変換されますが、DNS の設定が適切でなかったりすると、IP アドレスを得ることができず WWW サイトにアクセスができなくなります。

ネットワークの利用に際しては、このように重層的な仕掛けを利用していることを理解しておくと、トラブルの把握や相談、問題解決が容易になります。

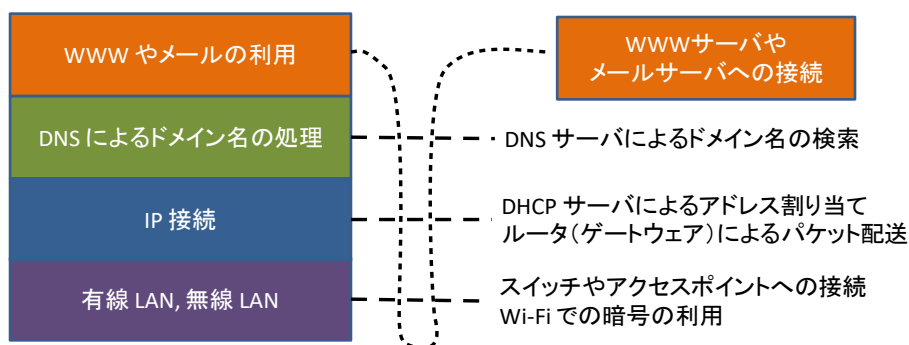


図 44 重層的なネットワーク接続とその利用

3.8 電子メールの利用

3.8.1 メールの確認

大学での学習や学生生活ではメールサービスは大学から皆さんへの重要な連絡手段です。日頃使っているメールサービスへの転送設定もできますので、定期的に（1日に一度程度は）メールの有無をチェックするようにしてください。夏休みなど休業期間中でも連絡を行うことは少なくありません。

3.8.2 安全な運用

1) 電子メールを悪用した攻撃

メールはウイルス感染や詐欺などにも悪用されます。電子メールではその設計上、差出人のアドレス（From アドレス）を容易に偽装できるということを理解しておくこともメッセージを確認する際に重要な知識です。

不審なメールの添付ファイルを開くことや、本文に示された Web サイトへのアクセスは行わないようにしてください。

- 標的型攻撃：近年では個人を特定した攻撃が多くなっています。このような攻撃では、添付されているウイルスも対策ソフトで根治することが難しくなっています。
- フィッシング：銀行などを詐称して偽のサイトに誘導し、口座の情報などを窃取するタイプの攻撃です。
- ランサムウェア：添付ファイルなどとしてウイルスを侵入させ、これによりファイルを勝手に暗号化して人質に取り身代金を要求する攻撃です。一度暗号化されるとそれを回復することは困難です。日頃からファイルをバックアップしておくことが重要な予防策です。

電子メールのアカウント（ID とパスワード）が他人に漏れると、あなたに成りすまして、迷惑行為を行うことができてしまいます。アカウントの管理はこういう意味でも重要です。

また、ストーキングのような行為として電子メールを何度も送りつけられるということも生じ得ます。このような被害に遭われた場合は、メールアドレスの変更などの対応が必要になります。

2) 不注意な操作で生じること

電子メールの操作では、以下のような誤りが生じることがあります。重要なメッ

セージ、他人からのメッセージの取扱いに注意しましょう。

- **送り先を間違ってしまう**：一度、送出すると取り消すことは困難です。重要なメッセージを扱う際には送り先を間違えて送ってしまわないように注意する必要があります。メーリングリストなどを利用している場合は多くの方に情報が流れるので注意が必要です。
- **送信者の了解を得ずに転送する**：電子メールでは転送が容易で便利ですが、送信者はあなただけが読むことを前提にメッセージを送っている場合があります。うっかりすると陰口など送信者が他者に送ることを望まないメールを転送してしまうことがあります。転送の際には送信者が了解していることなどを確認してください。

3.8.3 適切なメッセージの構成

1) フォーマルなコミュニケーションとしてのメール

教職員はメールによる皆さんからの連絡には様々な問題を感じています。これまでのメール利用（やその他のコミュニケーション手段）では

- 主に家族や友人との私信が中心に
- スマートフォンなどを使って短いメッセージを発信する

という使い方をされていたと思います。

これに対して大学でのメール利用は

- 大学での学習を含めさまざまな社会的活動で用いるフォーマルなコミュニケーションです
- メールアドレスだけで個人を特定することは困難です
- 連絡する相手（教員など）は数多くの業務を抱えています

このような状況で確実にメッセージが相手に届くことを心がけてください。

2) メッセージの構成で留意すべきこと

具体的には

- 件名を明記してください。
- 受取人を明記してください。（受取人は良く確かめて）
- 差出人の所属と氏名を明記してください。
- 用件を受取人が理解できるように書いてください。
- 適度な文字数で改行するようにしてください。

3) メールの宛先等の記載の仕方

メールの宛先等の記載の仕方ではメッセージの取り扱いが変わります。以下のことを知って利用してください。

- To: 主として読んで欲しい宛先として使います。
- Cc: 複写（カーボンコピー）を送っておく宛先として使います。To: や Cc: に書かれた他の受信者にもここに書かれたメールアドレスは見えるようにメッセージが作成されます。
- Bcc: ブラインドカーボンコピーの略です。Cc: と同じようにメッセージの複写を送りますが、他の受信者にはここに書かれた宛先は示されません。
- Reply-to: 返信をもらう際に From: に書かれた差出人宛ではなく別のアドレスに宛ててもらいたい場合に使います。

演習 13. メッセージについて

あなたがもし以下のピンク色のようなメールを受け取ったらどのような問題を感じるでしょうか？水色のメッセージだと何が良いでしょうか。

件名： レポート間に合いません 待ってください！
件名：情報基礎演習のレポートについて 〇〇先生 火曜4限の情報基礎演習を受講している 〇〇学部一回生の〇〇です。 〇月〇日締め切りのレポートですが、インフルエンザで寝込んでいたため、期限までの提出ができません、〇月〇日に提出したいと思いますがお認め頂けますでしょうか。

図 45 困ったメールのメッセージ

3.9 デスクトップの仮想化

さまざまなサービスを WWW で受けることが可能になりましたが、パーソナルコンピュータのウィンドウなどを使ったデスクトップ環境で行う作業も少なくありません。このようなデスクトップ環境をサーバ上に構成し、遠隔の端末から利用する方式をデスクトップの仮想化と呼び、そのための基盤を仮想デスクトップ基盤(VDI, Virtual Desktop Infrastructure) と呼びます。

このような仮想デスクトップ基盤を利用することには以下のような効果があります。

- 場所を選ばずに利用できる。
- データを持ち出さずにすむ。情報漏えいなどの防止に役立てることができます。
- 計算機の資源を共有できる。サーバ上で複数のユーザのデスクトップを稼働させることで、CPU などを効率よく使うことができます。
- パーソナルコンピュータのソフトウェアの新規導入や更新などの運用管理を管理者側で一元化できる。

他方、デスクトップ上の表示画面をネットワーク経由で手元の端末まで伝送するため、通信速度により描画の制限を受けたりして使い勝手がさがることがあります。

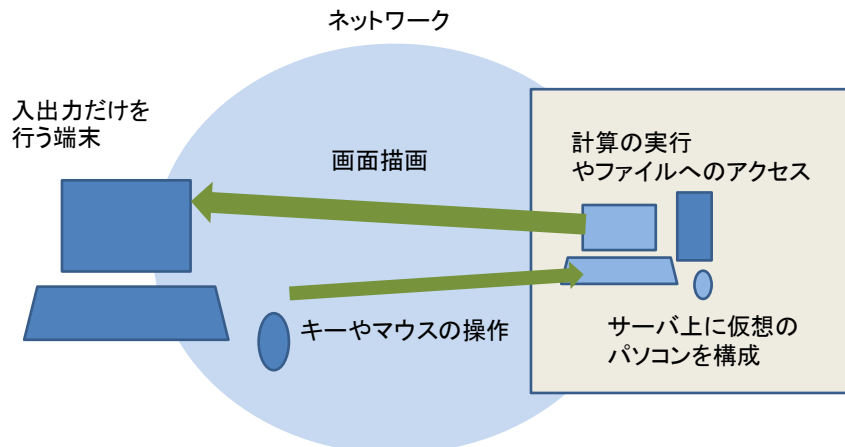


図 46 仮想デスクトップ基盤(VDI)

3.10 京都大学のネットワークサービス

3.10.1 京都大学のネットワーク KUINS

1) KUINS の概要

京都大学の学術情報ネットワーク KUINS (Kyoto University Integrated Network System)は以下の 3 系統で運用されています。

- **KUINS-II:** Web サーバなどを設置するための LAN で、グローバル IP アドレスが与えられます。
- **KUINS-III:** パーソナルコンピュータなどを接続するための LAN で、プライベート IP アドレスが与えられます。
- **KUINS-Air:** 学内の無線 LAN で KUINS-III と同様、プライベート IP アドレスが与えられます。KUINS-Air には多くの本学の学生教職員が機器を接続することから KUINS-Air に接続したノート型パーソナルコンピュータやスマートフォン相互の直接の通信は安全のため行えない運用をしています。

KUINS-III に接続したパーソナルコンピュータから外部の Web サーバなどにアクセスできるように中継の仕組みが KUINS-II と KUINS-III や KUINS-Air の間に設けられています。中継の方法として現在は透過型代理サーバ方式が用いられています。

2) 京都大学での無線 LAN アクセスポイントへの接続

大学の Wi-Fi アクセスポイントでは幾つかの方式が併用されています。

- **KUINS-Air:** ECS-ID, SPS-ID で接続します。学内限定の Web サイトなども閲覧できます。
- **eduroam:** 大学間で無線 LAN を相互に利用できるサービスです。eduroam 専用のアカウントを KUINS から取得して使ってください。eduroam に参加している他大学でも使えます。
- **公衆無線 LAN:** NTT ドコモ, KDDI(au), SoftBank が提供している無線 LAN サービスを学内のアクセスポイントで使えます。

3) 京都大学での VPN の利用

京大では学外から学内のネットワークに安全に接続するために IKEv2 方式などの VPN 接続サービスを提供しています。

3.10.2 京都大学のメールサービス

京都大学の学生にはメールサービスとして学生用のメールシステム KUMOI のアカウントが与えられます。また重要な情報が KULASIS や MyKULINE などからの連絡を含めて KUMOI に送られます。

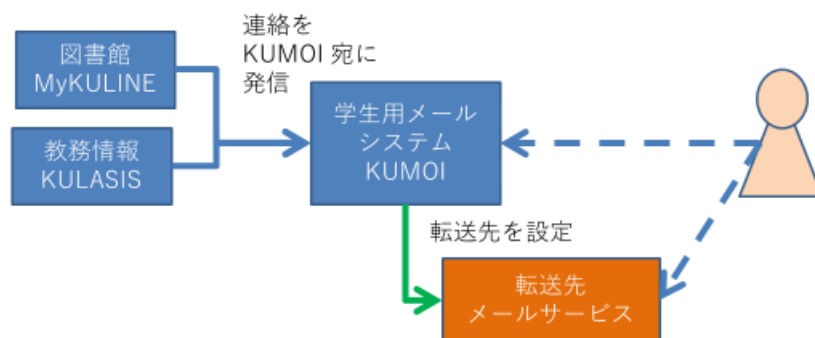


図 47 京都大学の学生向けメールサービス

3.10.3 京都大学の端末とストレージサービス

京都大学では学生の皆さんが利用可能な端末とストレージのサービスを教育用コンピュータシステムとして提供しています。

- **固定型端末サービス**： Windows 10 が利用できるパーソナルコンピュータ端末が演習室のほか自習用端末室 (OSL) に配置されています。学生，教職員用のアカウントをお持ちならどなたでも使えます。
- **仮想型端末**： Windows 10 と Linux が利用できる VDI 環境です。この環境を利用する授業ごとに利用を許可しています。
- **クラウドストレージサービス**： 教育用コンピュータシステムの端末環境からもネットワークに接続した個人の端末からも利用可能なストレージサービスです。他の利用者とのファイル共有も可能です。

3.11 情報通信技術 (ICT) の利用とリスク

1) 情報の持つ特性

情報には様々な人にとって**商業的価値**や**プライバシー**など様々な価値があります。一方、デジタル化されたネットワーク上で扱われる情報には以下のような特性があります

- 複製が極めて容易です
- 一度、社会に広がると消すことは困難です
- 一度消えると戻りません
- それだけでは真偽を判断しにくいです

2) ICT利用が孕むリスクが問題になる理由

インターネットで世界中のコンピュータが繋がり、それを様々な人々が利用していることから ICT の利用はさまざまなリスクを孕んでいますが、その理由は

- ネットワークが世界規模で広がり、高速な通信と情報処理が行われており、
- 大量の情報が保持されています。
- そして様々な意図や価値観を持つ多数の人々が利用しています。

このため、大きな影響を持つ可能性があります。

リスクを無くすことが難しいのは以下のような理由があるからです。

- コンピュータのソフトウェアは複雑で脆弱性を完全には無くせません
- 利用する人は不注意などで適正でない行動をとってしまいます
- 悪意を持つ人を含め、さまざまな人がインターネットを使っています

3) 情報の何を守るべきか

情報セキュリティの管理では情報について以下の3つの視点で考えます。

- **機密性**：情報の機密性が失われてしまうリスク
- **完全性**：情報が不完全になってしまうリスク
- **可用性**：情報が利用できなくなってしまうリスク

4) リスクの要因

情報に関するリスクを分けると物理的なリスクと人の行動に伴うリスクがあります

- 物理的なリスク
 - 機器の故障、停電などによる停止

- 人の行動に伴うリスク
 - 判断や操作の間違い，サービスの不十分な理解
 - 悪意のある行為
 - ネットワーク上での私刑的な行為，風評被害など数多くの人々の反応

5) リスクの回避

上記のような情報の持つ特性とリスクについて理解し，情報の取扱いと情報機器の操作について適正な行動をとる必要があります。詳しくは情報セキュリティ e-Learning で学習してください。

演習 14. パーソナルコンピュータのネットワークインターフェイス

あなたがパーソナルコンピュータをお持ちなら以下について調査してください

- 有線 LAN のインターフェイスと対応する通信方式
- 無線 LAN のインターフェイスと対応する通信方式
- 自宅でパーソナルコンピュータをネットワークに接続しているなら，その接続方式，無線 LAN の場合は暗号化方式

演習 15. パーソナルコンピュータのアドレス

あなたのパーソナルコンピュータの有線 LAN や Wi-Fi に割り当てられている MAC アドレスを調査してください。

またあなたのパーソナルコンピュータがネットワークに接続されているなら，割り当てられている IP アドレスを調査してください。多くの場合，IP アドレスとしてプライベート IP アドレスが割り当てられているはずです。3 種類あるアドレス範囲のどれに該当するのか調べてください。

演習 16. 情報セキュリティ e-Learning の修了

本学が提供している情報セキュリティ e-Learning を受講し，修了テストを受けてください。

演習 17. パーソナルコンピュータの学内ネットワークへの接続

あなたがノート型のパーソナルコンピュータやスマートフォンをお持ちなら以下の実施手順を確認してください。

- パーソナルコンピュータを学内の無線 LAN (KUINS Air) に接続する。
- 自宅から学内ネットワークに VPN 接続する。

演習 18. Web の暗号化と証明書の確認

本学があなたに提供している KULASIS や PandA などの Web システムでは通信が暗号化されており，適切な証明書を有していることを確認してください。

演習 19. 適切な電子メールの利用

授業などで担当教員に電子メールで連絡する際に，適切なメッセージの構成について述べてください。

4. 知的生産について

4.1 本章の目的

本章ではレポート作成などの活動を知的生産ととらえます。この章では後につづく具体的なスキルを位置付ける前提として

- 知的生産とはどういうことか
- 具体的にどのような活動が求められているのか
- 知的生産を効果的、効率的に行うために ICT がどのように活用されるのか
- 知的生産やそのための ICT 活用スキルを身に着ける上で考えておくべきことは何か

を知って頂くことです。

4.2 知的生産とは

「知的生産」とは民族学者だった梅棹忠夫氏の造語です。同氏の著書「知的生産の技術」では以下のように説明されています。

ここで知的生産とよんでいるのは、人間の知的活動が、なにかあたらしい情報の生産にむけられているような場合である、とかんがえていいであろう。この場合、情報というのは、なんでもいい。知恵、思想、かんがえ、報道、叙述、そのほか、十分ひろく解釈しておいていい。

(梅棹 忠夫: 知的生産の技術 (岩波新書) 岩波書店 (1969))¹²

すなわち、現代社会では多数の人は主に「知的生産」に携わっているのです。

「生産」という言葉で連想されるのは農林水産業や製造業で食料、エネルギー、住居、衣類、機械などを生産することでしょう。しかし現代の社会では多くの人は直接、物的生産に関わりません。図 48 は職業別就業者の構成割合の推移です。この図を見ると農作業などに関わっている就業者はすでになんかなり少なく、製造業などで直接、生産工程などに関わっている就業者も減少傾向にあることが分かります。こ

¹²この本は名著ですが、書かれたのは 1969 年です。パーソナルコンピュータもインターネットも携帯電話もまだない時代に書かれています。著者は民族学(文化人類学)が専門で、フィールドでの調査をまとめて論文、著書にするというのが仕事の形態です。同書では著者の専門での仕事の仕方を背景に「知的生産」について書かれています。

れに対して赤色で囲った部分が知的生産，例えばオフィス業務，専門的業務，販売業務などに関わっている就業者に相当し，すでに就業者の中で大きな割合を占めるとともに，増加しつつあることが分かります。

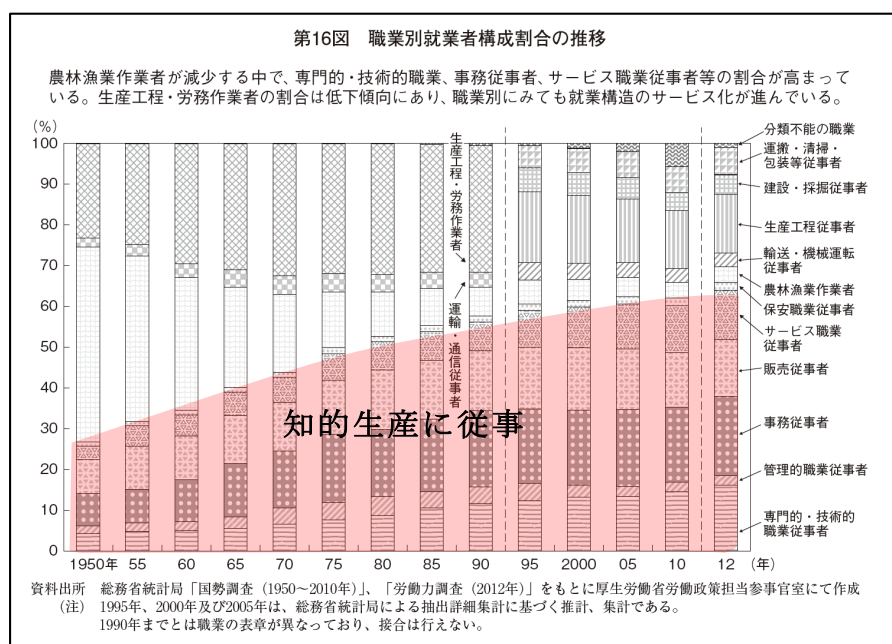


図 48 職業別就業者の構成割合の推移

平成 25 年版 労働経済の分析＜要約版＞－構造変化の中での雇用・人材と働き方－，
図 1 6 に加筆

<http://www.mhlw.go.jp/wp/hakusyo/roudou/13/dl/13-2-4.pdf> (2017/1/5 アクセス)

4.3 学園祭の活動と知的生産

学生生活に引きつけて知的生産について具体的に考えてみるために学園祭に模擬店を出店することを考えてみましょう。学園祭で模擬店を出すにあたって必要な作業などを書き出してみるとさまざまなものがあります。

- 模擬店を出す仲間作り
- 学園祭の実行委員会の出店条件などの確認
- 何を出店するかの検討
- 企画書の作成と提出，許可をもらう
- 機材類の手配，
- 安全衛生管理の理解，実施体制の検討
- スタッフの役割分担

- 予算，チケットの価格決定，印刷，販売
- 実施
- 決算，意見集約と反省
- 報告書の提出

ずいぶん，多くの作業が伴いますが，このうち，太字の箇所は，実際に「もの」を動かしたり，加工したりする作業は殆どなく，打ち合わせをしたり，調査や手配をしたり，売り上げや支出を想定して予算など立てたり，文書を作成したり，といったもので，ほとんど「情報」だけを扱う「知的生産」になります。

知的生産をいかに効果的（役に立つ），効率的（手間をかけない）に行うかが重要になりますし，知的生産が情報を扱うものですから，そのためにいかにパーソナルコンピュータやインターネットなどの情報通信技術(ICT)を使うかが重要になるのです。

4.4 大学生生活と知的生産

先の例では学園祭での模擬店出店で多くの知的生産活動が伴うことを見ましたが，大学生活では様々な知的生産活動を行わなければなりません。

● その代表例は**授業で課されるレポート**であるとか，**卒業論文**などの作成です。このほかにも

- さまざまな目的での**文献調査**や**アンケート調査**などを行い，その**報告書**をまとめることがあると思います。
- イベントやプロジェクトなどに関わると
 - その実施を主張するための**企画書**の作成
 - 具体的に実施するための**実施計画書**の作成
 - イベントでの**アンケート調査**のための質問紙の設計
 - 活動を広く伝えるための**広報物**の作成
 - 活動を行った後の**実施報告書**の作成
 - **実施報告の評価**

などが行われるでしょう

- またソフトウェア（コンピュータのプログラム）や情報システムなどを作ると
 - システムが何を実現すべきかを書いた仕様書
 - システムがどう設計されているかを書いた設計書
 - プログラムそのもの

➤ 操作マニュアル

➤ Q & A 集,

などの作成が必要になります。

最近では文書などの報告書とともに,

● スライドを作成して**プレゼンテーション**を行うことが多く求められるようになっていきます。また、これらの活動をチームで行う場合には

● 会議資料や議事録の作成が求められますし,

● 会議を上手に運用することや,

● プロジェクトの進捗を管理するプロジェクトマネジメント的手法の適用も必要になります。

4.5 知的生産を構成する活動

知的生産を構成する活動にはいくつかあります。

● 調査

➤ 文献調査

➤ フィールド調査, アンケートやインタビューあるいは観察

➤ 実験

➤ コンピュータシミュレーション

● 資料の整理・分析・発想

➤ ブレインストーミングや KJ 法などの発想法の活用

➤ 統計的分析やテキストマイニングなどのデータからの知識獲得

● 問題解決策の考案

● レポート作成

● スライド作成, ポスター作成とプレゼンテーション

● 会議の運用と議事録作成

● プロジェクトの管理

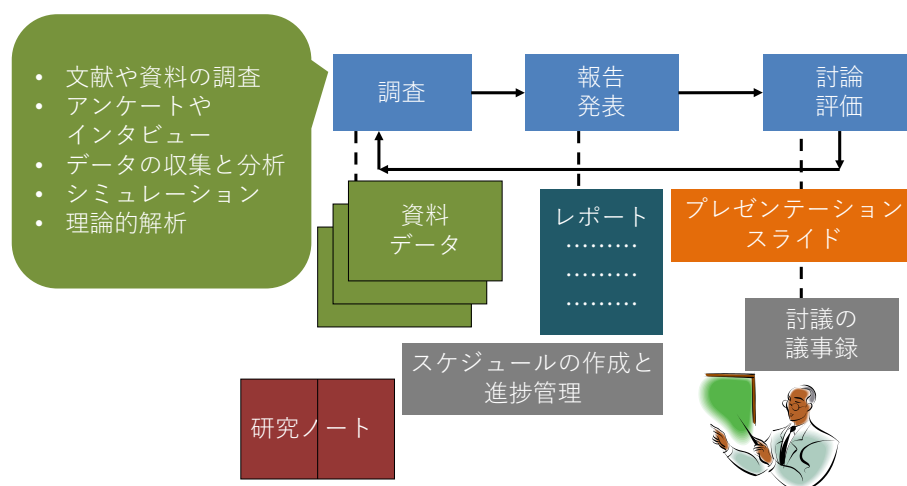


図 49 知的生産のプロセス

4.6 知的生産と ICT 活用

知的生産は主に情報を扱う仕事ですから、そのために ICT をいかに活用できるかは重要なことでしょう。授業での課題についてグループでレポートを作成し、プレゼンテーションすることを考えてみましょう。以下のように ICT を活用するでしょうし、そのためのスキルが重要であることが分かります。

- **文献調査**： 図書館での蔵書検索やデータベース検索を活用した資料調査や Web での情報検索
- **政府による統計などのデータ活用**： 政府や自治体などはさまざまな統計を Web で公開しています。そのデータを表計算ソフトを用いて分析したり、グラフに表したりすることが求められます。
- **資料整理とノート作り**： 文献などの資料，データをコンピュータ上のファイルとして整理したり，検索したりすることが求められます。また，読んだ文献の内容などは出典を記載するための書誌情報とともにノートのまとめておくことが必要でしょう。
- **コンピュータを用いたシミュレーション**： 表計算ソフトを用いたり，自分自身でプログラミングしたり，専用のソフトウェアを用いたりするなどしてシミュレーションのための数値計算を行います。
- **図表の作成**： データ分析やシミュレーションの結果を要約した表にまとめたり，グラフで表現することのほか，組織の構成などを図（ダイアグラム）にしたりします。
- **レポートの作成**

- スライド作成とプレゼンテーション
- グループでの活動： 情報共有，連絡，ミーティングと議事録の作成，レポートやプレゼンテーションスライドの校閲

4.7 知的生産スキルの獲得と学習の方略

4.7.1 スキル獲得の視点

以下の視点で知的生産スキルを獲得してください。

- あなたの知的生産活動は効果的な成果物を得ているでしょうか
- あなたの知的生産活動は効率的に行えているでしょうか
- あなたは知的生産のためのスキル獲得を継続的行えているでしょうか

4.7.2 目標の明確化

知的生産のために ICT スキルを獲得するに際しては目標を明確に意識する必要があります。すなわち

- 何 (What) をできるようになればいいか，成果物としてどのようなものを生み出すことを求められているかを意識します。
- どのようにして (How) 行うか。求められる成果を達成するために ICT をどのように使えばいいかを意識します。

参考文献

- [4] 梅棹 忠夫：知的生産の技術（岩波新書）岩波書店（1969）

5. 学術情報の探索

5.1 本章の目的

この章では、前章で学習した大学における知的生産に不可欠な学術情報に関して、1) 探索方法、2) 評価方法、3) 適切な活用方法の3点を学習します。大学における知的生産の中で、この章で想定しているのは、授業レポートや卒業論文など、皆さんの学習や研究の到達度を文章で表現する場合です。具体的な内容は以下のようになっています。

- 学術情報とは何かを理解する
- 検索のための概念整理のやり方を身につける
- 引用の仕方と参考文献リストの作成方法を身につける

これらを修得した上で、実習を通してオンライン文献検索の基礎を身につけましょう。

5.2 学術情報と図書館

5.2.1 学術情報とは何か

学術情報の特徴を説明する前に、情報化時代に大学で学習する内容について少し考えてみたいと思います。私たちの普段の生活では、分からないことがあればネットで検索しておおむね概要をつかむという習慣が定着しています。ネット検索では、ずらっと並んだ検索結果のそれぞれが書かれている文脈を考えるよりは、同じ単語を使っている、少しずつ違う文脈内書かれた内容を、なんとなくつなぎ合わせて考えることが多いかもしれません。一方で、大学での学習や研究をすすめる上では、文脈（コンテキスト）であるとか、各学問分野における枠組み（フレームワーク）の中で理解し、自分の意見や発見をその中に位置づけるという作業が重視されます。

もちろん、「東南アジアにおける、エビの養殖事業をめぐる流通と環境問題」といったように、ちょっと考えるだけでも、経済学、環境学、地域研究といったいくつかの分野にまたがる研究テーマはたくさんあります。グローバル化した現代社会における事象が研究対象であればそういった分野横断型の課題の方が多いでしょう。しかし、分野を超えたテーマにアプローチする基礎体力として、まずは自分の所属

する学部が対象とする学問分野の文脈と枠組みにのっとして考える習慣を身につけることが必要になります。文脈と枠組みというのは、スポーツに例えるとルールと競技会場といったところでしょうか。競泳と野球では、会場もルールも全く違いますが、こういった場で行うかが規定されていて、一定のルールに従って動き、先行事例を参考に戦い方を決めるといった点は共通しています。学問の場合でも、同じことがいえます。

このような前提に基づいて、学術情報を定義してみますと「特定の分野における、学問の進捗過程を、世の中に公開するためにつくられた情報」となり、具体的には、以下のような形態があげられます。

- 学会発表の内容
- 学術雑誌に投稿された論文
- 専門書
- 専門家によって執筆された百科事典や専門事典の項目

（百科事典といっても Wikipedia は誰でも書き換え可能なので学術情報ではありません）

そのような学術情報の集積の中で、先行事例、つまり先行研究を網羅的に検証した上で、自らの論を組み立てたり、新たな調査を行ったりするというのが大学における知的生産活動だといえます。

5.2.2 知識の定着と資料

学術情報として具体的にいくつかの種類 of 資料をあげましたが、知識の定着というのは、河の流れのようになっていて、多くの場合は、学会や学術雑誌で発表された最先端の研究内容が時間とともに定説化していきます。上流にあたる学会での発表や、研究論文として公開された新たな知見は、同じ分野の研究者の間で相互に評価・検証する「ピアレビュー（peer review）」というプロセスを経た上で、学術的に信頼できる情報としての評価が確立されます。その上で、学術書や専門書に掲載されることで、一般読者よりは限定されているかもしれませんが、近接分野の専門家にも知識として共有されるようになります。さらに下流に行くにしたがって、より一般的人に使いやすい、教科書や事典といった出版物や媒体に発信先が変化していき、その過程で残った知見は一般的な知識として定着していきます。

「iPS 細胞（人工多能性幹細胞）」を例にとって考えてみましょう。「iPS 細胞（人工多能性幹細胞）」は 2006 年に、*Cell* という学術雑誌に掲載された論文記事として世

に出ました。その時点では、専門家を中心としてごく限られた範囲で共有されていた知識にすぎませんでしたが、*Cell* に掲載した論文の著者である山中伸弥教授と、1960年代にその先行研究にあたるとされる論文を発表したイギリスのジョン・ガードン博士が、2012年にノーベル生理学・医学賞を受賞したことにより、多くの人々に知られる用語となり、現在では「iPS細胞（人工多能性幹細胞）」は、多数の一般書や百科事典などに掲載されています。

このような時間の経過と知識の定着の関係を、文献の種類とあわせて知っておくと、自分の知識レベルにあわせて学術情報の探索ができますので、頭の片隅にしまっておいて下さい。自分にとって全く新しいテーマに挑戦する時は、河の下流からさかのぼると、迷わずにたどり着けるはずです。

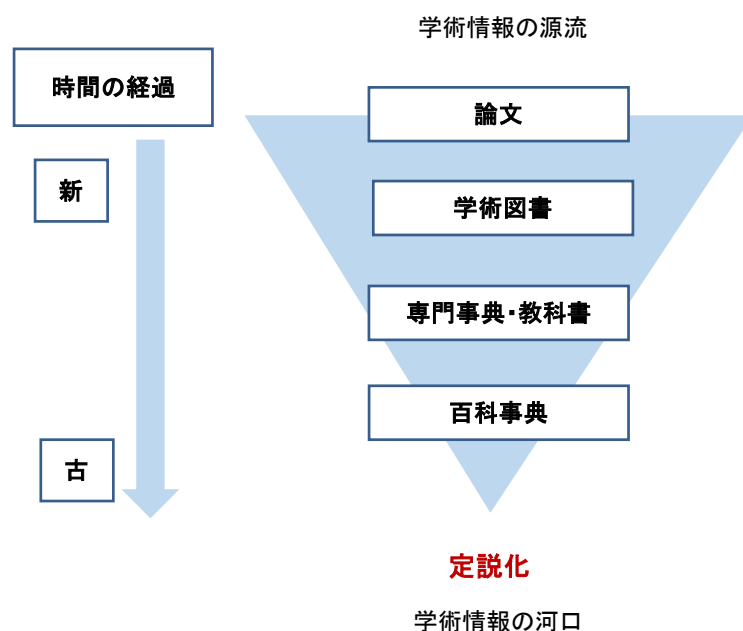


図 50 時間の経過と知識の定着

5.2.3 ネット上の学術情報と図書館が提供する学術情報の違い

さて、学術情報についてある程度理解したところで、ネット情報と図書館が提供する学術情報について簡単に整理しておきます。簡単にいうと、図書館経由で得られる学術情報は、「信頼性」が高いということです。あまたある情報の中で、図書館ではすでに定評がある学術出版物や、電子ジャーナルなどを購入し、提供しています。これは、専門家によるピアレビューを経た知識ということでもあります。一方で、ネット経由で手に入る学術情報は玉石混淆ですので、その中から本当に重要な学術情報を探し出すには、ある程度知識がないと難しいことが多いでしょう。

図書館では、以下のような「信頼性」の高い学術情報が提供されていますので、

これらを活用しない手はありません。なお、これらの図書館が提供している学術情報は、大変高価な資料で、個人で購入することが難しいものも多く含まれています。

- 参考資料（辞書・地図・統計など読みものではなく、調べるための資料を指します）
- 図書
- 学術雑誌
- 電子ブック
- 電子ジャーナル
- オンライン・データベース
- 新聞データベース
- 官公庁のレポートなど
- 機関リポジトリ

なお、新聞データベースや官公庁のレポートなどは、就職活動時や就職した後の調べ物にも役にたつ資料です。「信頼性」のフィルターをかけた資料を収集している図書館という場所で調べ物をはじめの習慣をつけておくと、学術情報以外にも、質の高い情報を選ぶ力が身についていきます。

5.3 検索を行う前に

5.3.1 文献検索の種類

学術情報と図書館については理解できたと思います。問題は、図書館という「信頼性」フィルターを通した後であっても、アクセスできる情報は桁違いに多いことです。このような情報の海の中で、網羅的な先行研究の収集を行うにはどうしたらよいのでしょうか。図書館における文献探索には、大きく分けて、1) ブラウジング、2) 芋づる式、3) オンライン検索の3種類があります。3つめのオンライン検索については、次節で詳しく説明しますが、3種類の文献探索方法を上手に組み合わせることが、不可欠です。

1) ブラウジング

- 図書館の棚をみながら探すことです。

- **メリット**⇒図書館では、分類方式に従って近い分野の本が同じ棚に並んでいますので、自分の考えているテーマを、より広く捉え直すことができますし、オンライン検索ではうまく探せなかった資料に出会えることも多々あります。
- **デメリット**⇒多分野にわたるテーマの場合、どこに分類されているかが分かりにくいことがあります。また、棚にあった本だけをみて網羅的な資料収集をしたような気になってしまわないように気をつけて下さい。

2) 芋づる式

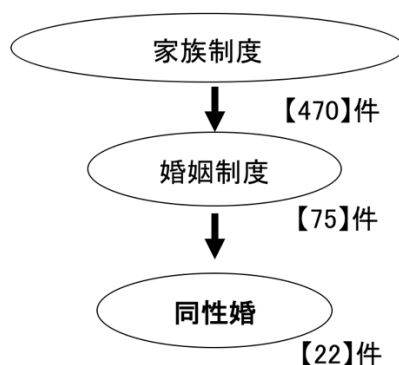
次節で説明する、先行研究の参考文献リストに挙げられている文献を探すことです。芋づる式を可能にする引用データベースというのもありますので、それに関しては章末のコラムを参照してください

メリット⇒自分の取り組んでいるテーマにおいて、重要な文献を効率良く入手できます。

デメリット⇒先行研究が言及していない重要な文献を見落としてしまうことがあります。

5.3.2 オンライン検索とキーワードの整理

次に、3つめの選択肢であるオンライン検索ですが、効果的なオンライン検索を行うために、まず必要な作業とはなんでしょうか。それは、自分が考えているテーマをキーワード化し、それらのキーワードの上位語、等位語、下位語を考えることで、検索キーワードをめぐる用語や概念を整理することです。ブラウジングや芋づる式と違い、オンライン検索の場合は、入力したキーワードによって検索結果が劇的に変わってきます。



【 】内は2018年12月17日のKuline検索結果

図 51 検索語と検索結果数

「現代社会におけるマイノリティと法」というレポート課題が出たとしましょう。現代社会とは、どの範囲を指すのか、どのようなマイノリティを取り上げたいのか、どのような法を考えたいのか、を整理していく中で、「同性婚」というテーマについて調べようと考えたとします。ところが、「同性婚」というキーワードだけで検索していても、なかなか思うような文献に出会わないかもしれません。そういった時は、検索しているキーワードの上位概念や、関連語を考えてみましょう。また逆に、自分の思っている検索語では、ヒットする文献の数が多すぎてどれを読んだらよいのか分からないということあります。そういう場合は、自分が考えた検索語の下位語を考えてみましょう。

図 52 の左側は、「大気汚染」というキーワードで、国立国会図書館の文献データベース『国立国会図書館サーチ』（URL: <http://iss.ndl.go.jp/>, 2018 年 12 月 17 日時点）を検索した場合に提示される、上位語・下位語・関連語です。「大気汚染」の上位語である「公害」で検索をするとしたら、「大気汚染」を含んだ、さまざまな「公害」に関する文献がヒットしますので、検索結果数は増加しますし、下位語の「光化学スモッグ」で検索した場合、「大気汚染」の中でも「光化学スモッグ」に言及している文献しかヒットしないので、検索結果数は減ります。

このように、文献を探索する際には、検索概念のレベルを行き来しながら、模索することが必要になります。オンライン検索を行っているうちに、他にも有効なキーワードが見つかることは多々ありますので、どの検索ワードによって、どのような文献を入手できたか、文献とキーワードを精査することで、自分が必要な情報を見つけていきましょう。このようにして入手した文献の中で、同じテーマを扱っている論文が高い頻度で引用している文献は、学術情報の上流に位置する文献、つま

り「基礎文献」である確率が高いですので、最新の研究成果と基礎文献の双方を活用する習慣をつけましょう。

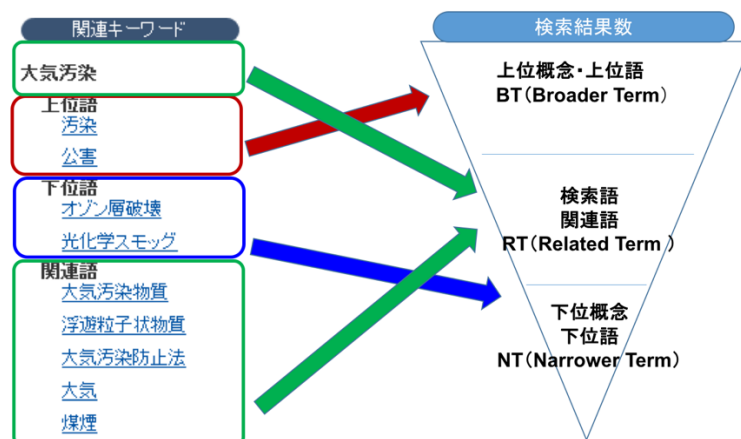


図 52 上位概念・下位概念

5.4 参考文献リストの読み方・書き方

もう一つ、文献検索をはじめる前に知っておく必要があるのが、参考文献リストの読み方と書き方です。参考文献リストとは、論文や本の最後に掲載されており、著者が引用した文献を一覧にしたものです。これを見ることによって、著者がどのような先行研究を参考にし、その上で、どのような新しい発見を提示しているのかが一目瞭然となります。前節で、芋づる式の文献探索方法について説明しましたが、レポートや論文を書く作業は、参考文献リストを読むことから始まり、参考文献リストを書くことで終わるといっても過言ではありません。なお、参考文献を示すことの重要性については、科学技術振興機構（2011）の『参考文献の役割と書き方』で、以下の5つが挙げられています。

1. 自身の論文の新規性、独創性、信頼性の明確化
2. 先行する著者（先人・先輩）に対する敬意
3. 出典の明示
4. 読者に対する情報提供
5. 著作権法の遵守

引用と参考文献リストを読んだり、作ったりする上で、もう一つ重要なのが、決められた書式に従うということです。フォーマットは、専門分野によって異なりますので、レポートを提出する場合は、事前に先生と確認しましょう。

引用・参考文献の代表的な書式

- APA (American Psychological Association: アメリカ心理学会)
【心理学をはじめ, 社会科学分野全般】
- Chicago 【人文学 (歴史・語学)・社会科学分野】
- MLA (Modern Language Association of America 米国現代語学文学協会)
【人文学 (文学・言語学・哲学)】
- IEEE (The Institute of Electrical and Electronics Engineers: 電気電子学会)
【電気・通信・電子・情報工学分野】
- SIST (Standards for Information of Science and Technology: 科学技術情報流通技術基準)

本文で引用した資料は, 必ず参考文献リストに挙げる必要があります。書式の多くが, 引用と参考文献のどちらも指定していますので, その場合は, どちらも同じ書式に従って記載しましょう。

英文の書式による記述

- APA
(本文中) (Klaus, 1981)
(参考文献リスト) B. Klaus. (1981). Robot Vision. Cambridge, MA: Little, MIT Press.
- ◆ IEEE
(本文中) as known by Klaus [4]
(参考文献リスト) [4] B. Klaus, Robot Vision. Cambridge, MA: MIT Press, 1981.

和文雑誌の書式による記述例

- 『社会学評論』
(本文中) (西川編 2001)
(参考文献リスト) 西川潤編, 2001, 『アジアの内発的発展』藤原書店。
- 『東南アジア—歴史と文化—』
(本文中) [生田・松澤 2000]
(参考文献リスト) 生田真人・松澤俊雄編. 2000. 『アジアの大都市 3 クアラルンプール, シンガポール』日本評論社。

また, 文献リストを作成する場合は, 同じ書式でも, 文献の種類によって形式が違います。特に指定がない場合は, SIST 書式を使用しましょう。

SIST による文献リスト

(以下の例では，便宜上分かりやすい箇所で改行しています)

● 図書の場合

◆ 単著図書の全体を参考にした場合

山崎茂明. 論文投稿のインフォマティクス. 中外医学社, 2003, 167p.

著者名. 書名. (版表示,) 出版者, 出版年, 総ページ数.

◆ 図書の一部を参考にした場合

山崎茂明. “国際ルールへの理解”. 論文投稿のインフォマティクス.
東京, 中外医学社, 2003, p.71-101.

著者名.“章の見出”. 書名. (版表示.)

出版地, 出版者, 出版年, 始頁-終頁.

● 雑誌論文の場合

◆ 著者が1名の和文雑誌記事を参考にした場合

根本信乃. 寿命関連遺伝子と「老化の代謝説」. 実験医学.

2003, vol.24, no.11, p.1624-1629.

著者名. 論文名. 雑誌名.

出版年, 巻数, 号数, 始頁-終頁.

◆ 著者が複数名の和文オンライン・ジャーナルを参考にした場合

荒川正幹ほか. Hopfield Neural Network を用いた新しい分子重ね合わせ手法の
3D-QSAR への応用. Journal of Computer Aided Chemistry, 2002,
vol.3, p.63-72. [http://joi.jlc.jst.go.jp/JST.JSTAGE/jcac/3.63,\(参照2016-12-03\).](http://joi.jlc.jst.go.jp/JST.JSTAGE/jcac/3.63,(参照2016-12-03).)

著者名. 論文名.

誌名, 出版年,

巻数, (号数,) 始頁-終頁. 入手先, (入手日付).

● ウェブページ内の記事の場合

中央教育審議会. “教育振興基本計画について－「教育立国」の実現に向けて
－（答申）”. 文部科学省. 2008-04-18.

http://www.mext.go.jp/b_menu/shingi/chukyo/chukyo0/toushin/08042205.htm, (参照
2008-08-13).

著者名. “ウェブページの題名”. ウェブサイトの名称. 更新日付.

入手先, (入手日付)

5.5 検索の実際

では、実際に図書館が提供している学術情報を検索してみましょう。その際、入り口となるのが Kuline（クライン）です。Kuline では、5.2.3 で挙げた、京都大学内の学術情報の検索や、代表的なデータベースへのリンクが提供されています。以下に Kuline へのリンクと、Kuline からアクセスできる、いくつか代表的なデータベースを挙げておきます。なお、京都大学がお金を払って契約しているデータベースの多くは、「同時アクセス数」が限定されています。データベースの使用後は、速やかにログアウトして下さい。

● Kuline	京都大学蔵書検索 【 http://kuline.kulib.kyoto-u.ac.jp/ 】
● Cinii Articles	日本語論文検索（全分野）
● JapanKnowledge Lib	多数の辞書や百科事典の検索
● 医中誌 Web	医学系論文検索（日本語）
● Pubmed	医学系論文検索（英語）
● Web of Science	英語論文検索（全分野）
● 聞蔵Ⅱビジュアル	朝日新聞・AERA・知恵蔵など
● 日経テレコン	日経4紙・京都新聞・日経会社プロフィール、財務情報、各種統計データなど

5.6 困ったら図書館に

本章では、学術情報の探し方と扱いについて概観してきました。このようなステップを経て探しても、学内で探している資料の所蔵がない場合は、図書館ネットワークを経由して、他大学からコピーや現物を取り寄せることができます。その他にも、分からないことがあれば、まずは図書館に相談してみて下さい。なお、附属図書館では、1階ラーニングコモンズにて、平日13時～19時の間、大学院生のスタッフに学術情報の探し方やレポートの書き方などの相談をすることができる「学習サポートデスク」が設置されています。

参考文献

文献探索についてもっと知りたい方は、以下の図書[5][6]もおすすめてです。

- [5] 井上眞琴. 図書館に訊け. ちくま書房, 2004, 256p.

- [6] 市古みどり編. アカデミック・スキルズ 資料検索入門—レポート・論文を書くために. 慶應義塾大学出版会, 2014, 151 p.
- [7] 独立行政法人科学技術振興機構. 参考文献の役割と書き方. 独立行政法人科学技術振興機構, 2011, 24p. jipsti.jst.go.jp/sist/pdf/SIST_booklet2011.pdf, (参照 2016-10-10).

Great Idea! ガーフィールド博士と引用文献データベース

現在、国際的な大学評価で使われている指標の一つに、評価対象の大学に所属する教員が書いた論文が、どのような雑誌に掲載され、どの程度の頻度で引用されているか（被引用件数）といった論文の影響力を数値化し、被引用文献から引用元をたどることができる引用文献データベースがあります。

つまり、①ひとつの論文からその研究の上流にある論文をたどり、源流にたどり着くことができますし、②その逆にある論文から、その後の研究がどのように展開していったかを調べることができるデータベースです。①の使い方をすると、その分野における「基礎文献」が分かります。

このような引用文献データベースの代表的なものに、Web of Science があります。Web of Science のはじまりは、*Science Citation Index* という冊子で、1961 年にはじめて出版されました。この冊子を出版したのが、当時 30 代だったユージン・ガーフィールド（Eugene Garfield）です。ガーフィールドは、すべての論文は、それ以前に書かれた別の論文を引用・参照していることから、引用索引を作り、論文間の関係性を明らかにすることで、科学の構造化を実現できるということに気がつき、1955 年に論文で発表しました。その後、全米の州裁判所・連邦裁判所の判例を索引から引き、参照することができる冊子、*Shepard's Citation* を参考に出版したのが、*Science Citation Index* でした。現在では、自然科学のみならず、社会科学分野の論文も集積され、Web of Science というオンライン・データベースで提供されています。

Eugene Garfield. Citation Indexes for Science- New Dimension in Documentation through Association of Ideas. SCIENCE. 1955, vol. 122, no. 3159, p,108-111.

6. 文献管理ツールを活用しよう

6.1 学習の目的

「5.4 参考文献リストの読み方・書き方」と「11.9 参考文献の記載方法とその引用」で、レポートや論文執筆のために収集した文献のうち、実際に引用した文献を「参考文献リスト」として記載することの重要性について学習しました。同様に、「参考文献リスト」の作成にあたっては、指定されたフォーマットに沿って形式を整える必要性についても学びました。

しかし、執筆するレポートや論文の数が増えてきたり、修正を繰り返したりしていると、うっかりと引用した文献を「参考文献リスト」に記載し忘れてしまうことがあります。さらに、テーマが近いレポートや論文では、以前に引用した文献を再度引用することがありますが、その際に、どのレポートや論文で使ったがすぐに分からなかったり、転記の際に間違えてしまったりすることもあります。さらに、指定されるフォーマットが、その都度異なるということもあるでしょう。

この章では、文献の収集から参考文献リストの作成までのプロセスをスムーズにする文献管理ツールについて概要を理解した後、文献管理ツールのひとつであるRefWorksを使用して、以下のプロセスを学びます。

- 様々な情報源から収集した文献情報をインポートし、自分の書誌情報データベースを作成する。
- 文献情報を特定の様式でエクスポートし、参考文献リストを作成する。

6.2 文献管理ツールとは？

文献管理ツールには、大きく分けて以下の二つの用途があります。

- 1) レポートや論文執筆のために収集した文献の書誌情報をデータベース化することで、蓄積・管理する
- 2) 1) から必要な情報を抽出して、参考文献リストを簡単に作成する

つまり、文献そのものというよりは、書誌情報の管理と活用を目的としているの

が文献管理ツールです。文献管理ツールを利用することによって、文献の収集から論文執筆、参考文献リストの作成までの流れをよりスムーズに行うことが可能になります。図 53 は文献管理ツールを使って、文献収集から参考文献リスト作成までを行うプロセスを示した図です。

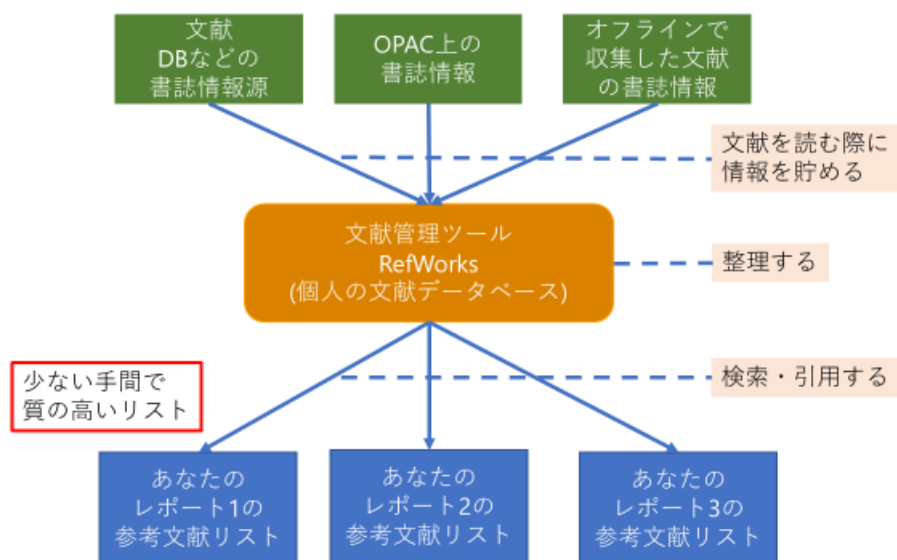


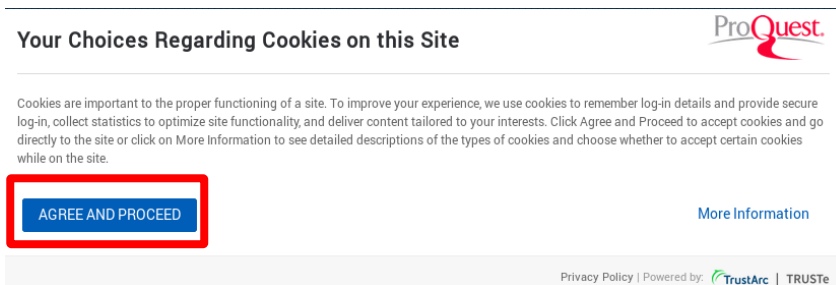
図 53 文献収集から参考文献リスト作成までを行うプロセス

6.3 RefWorks を使ってみる

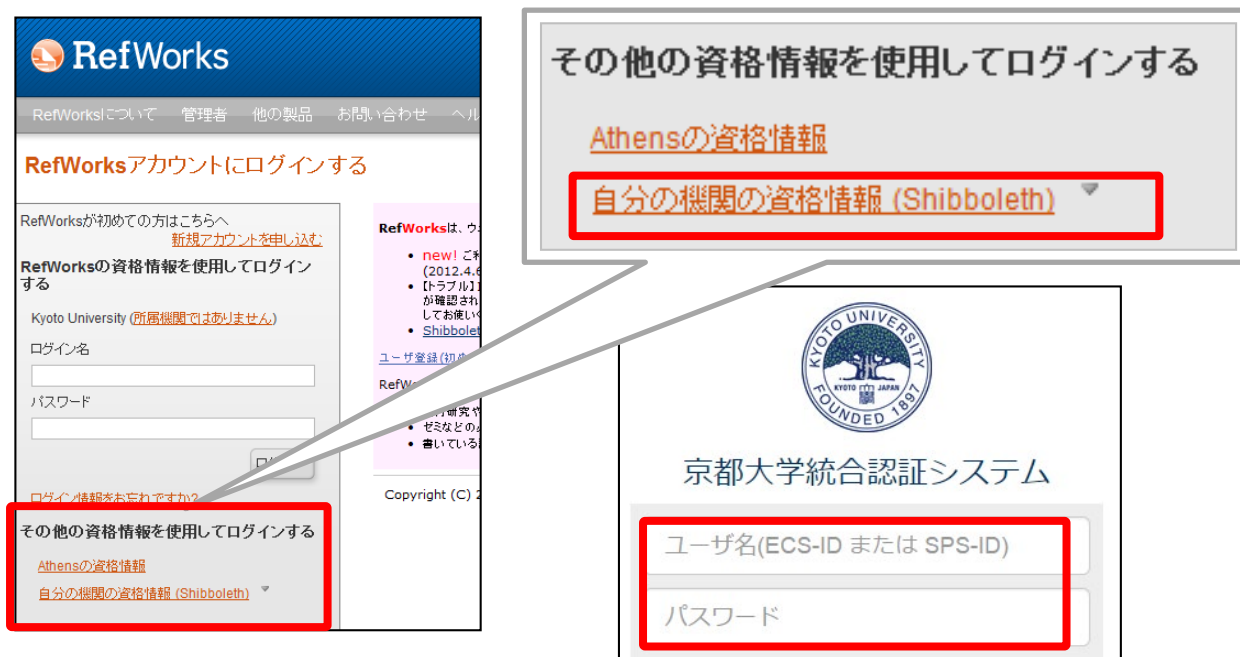
6.3.1 RefWorks のアカウントを作成する

文献管理ツールには、RefWorks、EndNote などの有料ソフト、Mendeley などの無料ソフトなど様々なものがありますが[8]、この章では京都大学で契約している RefWorks を使用して、作業を行ってみましょう。RefWorks を使用するには、まず事前にアカウントを作成する必要があります。次のステップで RefWorks にアクセスし、自分の情報を入力してアカウントを作成します。

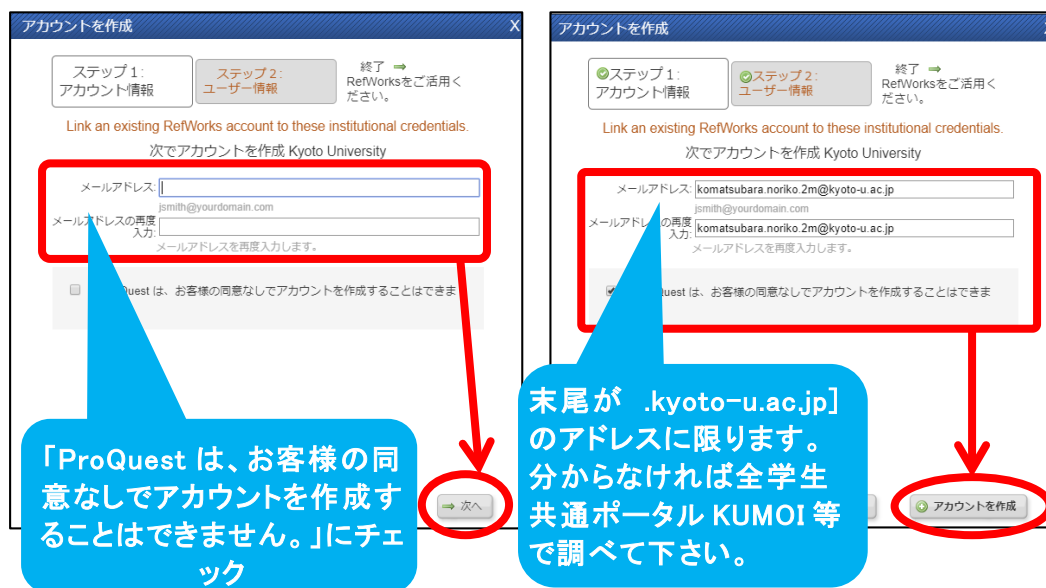
1. RefWorks に京都大学図書館機構サイトや KULINE 等からアクセスしてください。
2. 「Your Choices Regarding Cookies on this Site」画面がポップアップしますので、「AGREE AND PROCEED」をクリックしてください。



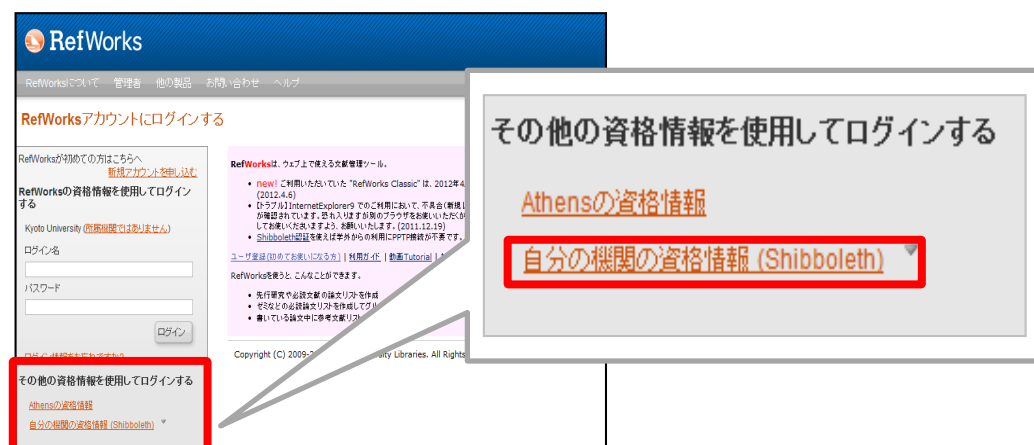
3. 「自分の機関の資格情報(Shibboleth)」から「Kyoto University」を選択します。
4. 統合認証システムの画面が現れたら ECS-ID とパスワードを入力してください。
別途、認証が済んでいる場合は画面が現れませんので次の作業に進んでください。
なお、学内で登録作業を行う場合は、「RefWorks 新規ユーザ登録にアクセスする」のリンクをクリックすると、自動的に Shibboleth 認証画面に移動します。



5. ユーザー情報画面で、末尾が[.kyoto-u.ac.jp]の自分のメールアドレスを入力し、必要なユーザー情報を入力します。



6. アカウントが作成出来たら、ログインしてみましょう。「自分の機関の資格情報 (Shibboleth)」から「Kyoto University」を選択してください。なお、RefWorks には、新旧 2 種類のバージョンがありますが、京都大学が契約しているのは、オレンジのアイコンで示される旧バージョンです。



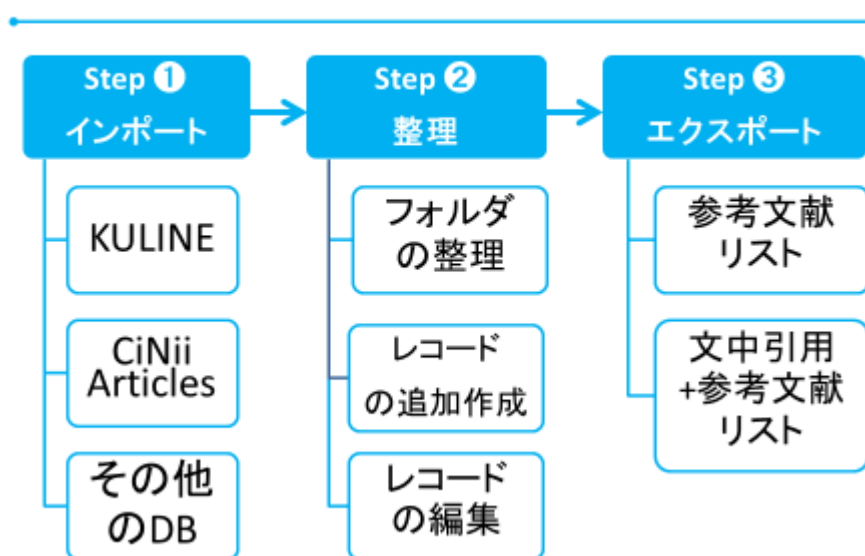
7. ログインボタンをクリックすると、クッキーについての確認画面が表示されるので、「AGREE AND PROCEED」をクリックしてください。

8. アカウント作成時に上記で入力したメールアドレスに登録内容が送付されますので、大切に保管してください。RefWorks を使っていて、質問がある場合は、以下のページを確認し、解決しなければ同じページ内の問い合わせフォームを活用して、メールを送信してください。<https://www.kulib.kyoto-u.ac.jp/form/14106> ただし、緊急の場合は、実際に図書館に足を運んで質問しましょう。

6.3.2 RefWorks を活用する

それでは、実際に使ってみましょう。次の3つのステップを、演習を通して行ってみます。

RefWorks 活用 3 Steps



RefWorks の画面は、以下のような構成になっています。

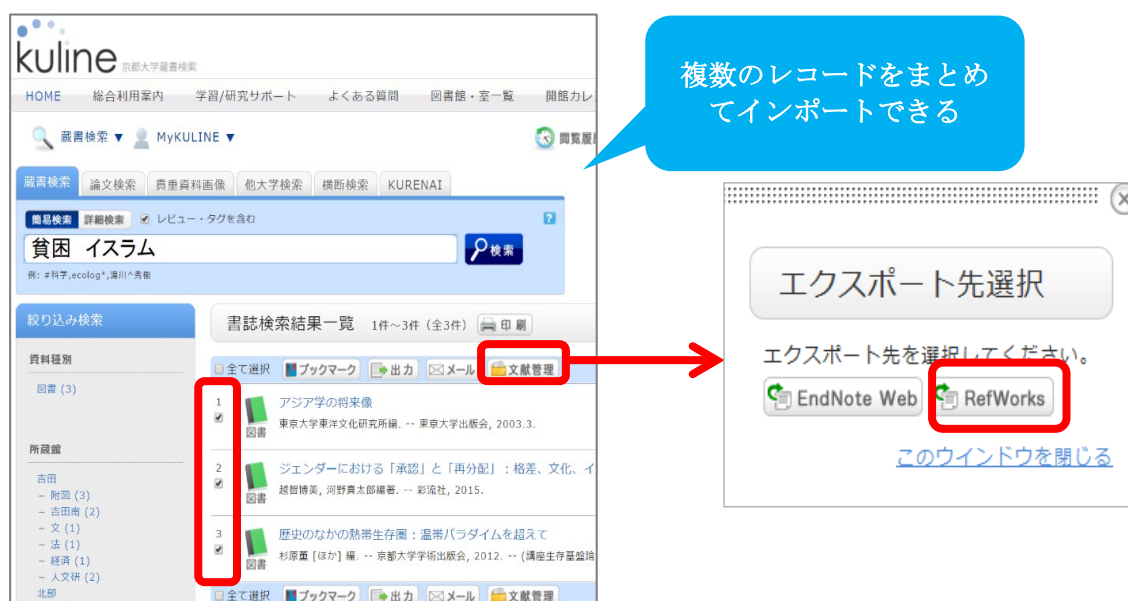


1) ステップ1 DB 検索結果から書誌情報をインポート

その1 KULINE から書誌情報をインポートする

でははじめに、皆さんが日常的に使っている図書館の蔵書 DB の検索結果から書誌情報をインポートしてみましょう。

1. KULINE 【<https://kuline.kulib.kyoto-u.ac.jp/>】 にアクセスし、各自のレポートのテーマや調べたい事柄に関する資料を検索しましょう。
2. 検索結果画面で、必要な書誌情報を選択し、☒をつけたのち、「文献管理」ボタンをクリックし、「エクスポート先を選択」という画面が出たら、RefWorks を選択します。



3. 書誌詳細画面でも、同様の作業を行うことで、RefWorks にインポートすることができます。



エクスポートの際に、以下の画面が出たら、左側の「古い RefWorks にエクスポート」を選択してください。



その2 CiiNii Articles から書誌情報をインポートする

RefWorks では、KULINE と同じようなやり方で、外部の書誌データベースからインポートをすることができます。

1. CiiNii Articles に [【https://ci.nii.ac.jp/】](https://ci.nii.ac.jp/) にアクセスし、各自のレポートのテーマや

調べたい事柄に関する資料を検索しましょう。

2. 検索結果一覧からインポートしたい論文に☑をつけ、「RefWorks に書き出し」を選び、「実行」をクリックしましょう。なお、タイトルの一部(<>の部分)がインポートされない場合があるので注意してください。



3. KULINE 同様、書誌の詳細ページからインポートすることもできます。この際、書誌情報の下にある「京大 RefWorks」ボタンを押してインポートすると、共著者情報が取り込まれませんので、右側のサイドメニューから「RefWorks に書き出し」を選ぶようにしてください。なお、CiiNii Articles の書誌情報のうち、翻訳などの別タイトルは、エクスポートされません。



その3 手持ちの文献や参考にしたウェブサイトの書誌を加える

RefWorks では、文献データベースからのインポートに加え、手持ちの文献や参考にしたウェブサイトの書誌情報を入力し、一括管理することができます。

1. 「レコードの作成」ボタンをクリックします。
2. フォーマットと、Reference Type（文献の種類）を選択し、手元にある文献を見ながら、書誌情報を入力してください。
3. Reference Type を新聞やオンライン・ジャーナルなど、参考にしたウェブサイトの情報を入力する場合は、「Reference Type」のプルダウンメニューから「ウェブサイト」を選びます。
4. 一通り入力したら、必ず「その他」をクリックし、新しくでてきたフィールドを可能な限り詳しく記入しましょう。

新規作成

選択済みの出力フォーマット: SIST02_2009.04 [このフォーマットについて](#)

and Reference Type: 書籍

以下の項目は選択した出力フォーマットで必須です。その他の項目は画面下の「その他の項目」からアクセスできます。

著者: 櫻井義秀

書名: 現代中国の宗教変動とアジアのキリスト教

出版年: 2017

総ページ数: 453

終了ページ番号、他:

版:

出版社: 北海道大学出版会

フォルダに追加: アジアの宗教

添付ファイル: 参照... ファイルが選択されていません。

その他の項目

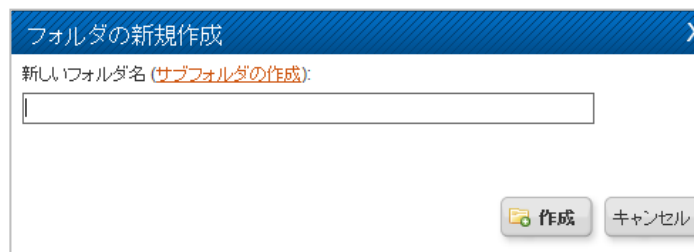
保存する 保存 & 新規追加

2) ステップ2 フォルダの整理

次に、RefWorks 内に蓄積した書誌を整理しましょう。「2.5.5 記憶装置とファイルシステム」で、フォルダを活用した階層構造によるファイルの管理について学習しましたが、RefWorks でも同様の考え方で、フォルダやサブフォルダを作成し、ディレクトリでファイルを管理するように、書誌レコードをフォルダ内で管理することができます。一つの書誌レコードを複数のフォルダ内に収納することが可能です。

で、授業課題ごとや調査テーマごとなど、使いやすいようにフォルダを作成し、必要に応じて複数のフォルダに書誌レコードを追加しましょう。

1. 「フォルダの作成」をクリックし、調査テーマにあった名前でも新規フォルダを作成しましょう。



2. レコード一覧で、フォルダに追加するレコードに ☒ します。
3. アイコンをクリックし、をクリックし、追加先のフォルダを選択してください。



4. すべての書誌レコードを一覧したい場合は、レコードタブをクリックしましょう。また、フォルダに整理した書誌レコードを検索したい場合は、検索メニューのプルダウンメニューから検索方法を選び、検索を実行しましょう。

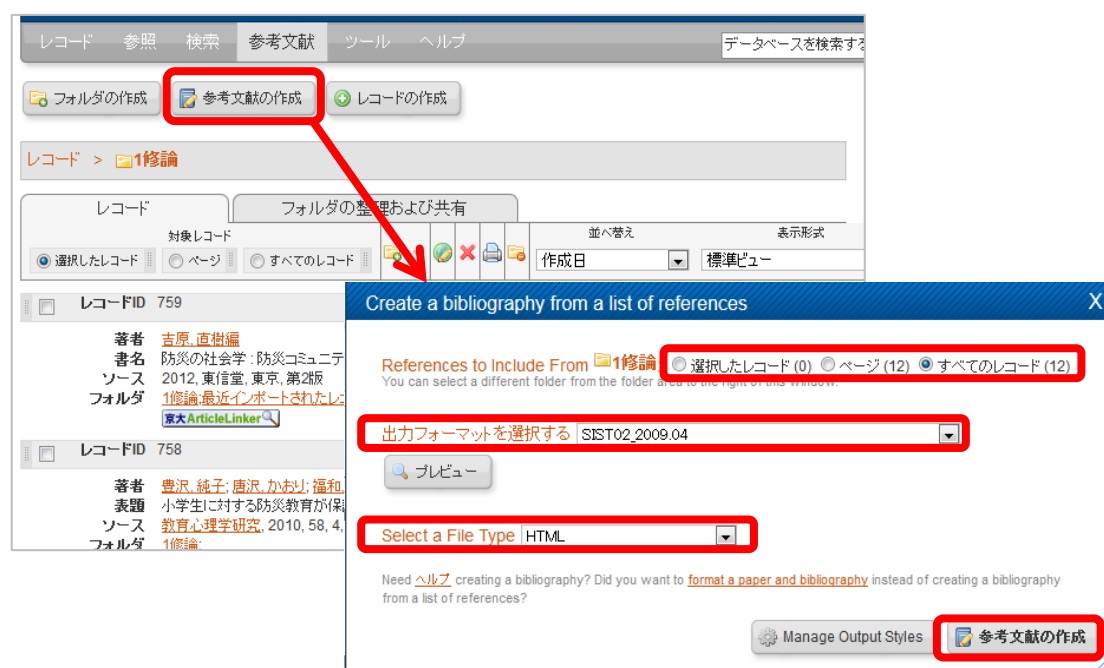
3) ステップ 3 参考文献リストと文中引用の作成

特に投稿論文の場合は、決められたフォーマットで「参考文献リスト」を作成することが非常に重要ですが、ピリオドや句読点の打ち方や、スペースの有無など、自分でタイプ打ちすると間違いが生じがちです。RefWorks では、自動的に指定のフォーマットにのっとり参考文献リストや文中引用を作成することができます。もちろん、自動生成ならではの間違いもありますので、最後に必ず確認しましょう。

参考文献リストの作成

収集した文献の中から、実際に使った文献を選択し、論文の末尾につける参考文献リストを作成してみましょう。

1. 参考文献リストを作成するフォルダを表示します。
2. メニューバーの下にある、「参考文献の作成」をクリックします。
3. 「出力フォーマット」を選択してください。出力フォーマットの種類に関しては、「5.4 参考文献リストの読み方・書き方」を参考にしてください。ここでは、出力フォーマットは、「SIST02_2019.4」を、ファイルタイプは「HTML」を選択しましょう。
4. RefWorks では、日本人の著者名が漢字コード順に並びます。日本語論文に関しては、最後に必ず自分で確認し、五十音順に並べなおしましょう。

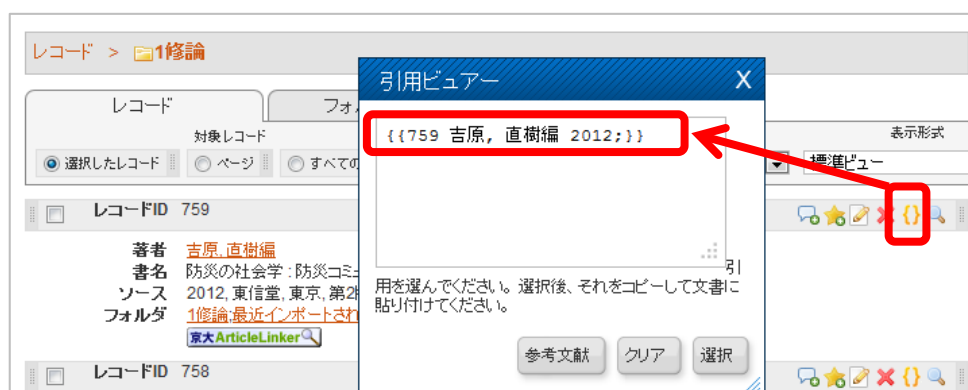


文中引用フォーマットの作成

参考文献リストと同じように重要なのが、レポートや論文の文中で引用した部分の後に、どの文献から引用したかを明示することです。文中引用は、参考文献リストと同じフォーマットで作成する必要があります。

レポートや論文の文中引用のフォーマットを自動生成してみましょう。

1. RefWorks で、 引用したいレコードの右上にある「{ }」のアイコンをクリックします。
2. 引用ビューアで「選択」ボタンをクリックし、反転部分をコピーし、レポートや論文の該当部分に、挿入しましょう。
3. 引続き、別の文献の文中引用フォーマットを自動生成する場合は、引用ビューア下の「クリア」ボタンをクリックしてから、上記の 1、 2 を繰り返しましょう。



演習 20. 調査したいテーマを設定し、RefWorks に KULINE と CiNii Articles からそれぞれ 3 書誌をインポートしてください。その上で、手持ちの文献を加え、「SIST02_2019.4」フォーマットで参考文献リストを作成しましょう。

4) RefWorks を活用するためのツール

RefWorks には、追加でダウンロードできるツールがあります。Write-N-Cite は、MS Word と連携し、RefWorks で管理している書誌データを、簡単に Word 文書で引用したり、参考文献リストを挿入したりするためのツールです。RefGrab-It は、ブラウザから直接、ウェブページの情報を RefWorks に取り込むツールですが、必ずしも正確に情報が入るわけではないので、取り込んだ後に編集をするか、ステップ 1 で学習したように、レコードを新規作成する方法をとみましょう。



【参考文献・ウェブサイト】

[8] Mendeley <https://www.mendeley.com/>

* 本章は、附属図書館定期講習会の資料を参考にしてしています。実習を繰り返して受けたい場合は、附属図書館の講習会に参加しましょう。講習会の詳細については、**【<https://www.kulib.kyoto-u.ac.jp/mainlib/service/workshops>】**を参照してください。

7. 表計算ソフトによるデータ処理（1）

7.1 学習の目的

- アカデミックな活動における表計算ソフトの利用目的を理解すること。
- 表計算ソフトの基本構造としてセルの内容と他のセルの値の参照法について学びます。
- 表計算ソフトへのデータの入力方法について、表の構成法と入力の制限法を学びます。
- 表計算ソフトに入力されたデータの特定の項目に注目した並べ替え方法を学びます。
- 表計算ソフトに入力されたデータの集計法について学びます。
- ピボットテーブルを用いたデータのクロス集計法について学びます。

7.2 表計算ソフトの利用目的

アカデミックな ICT スキルとして表計算ソフトを利用する目的についてここでは以下の3つを考えます。

- 大量のデータを適切に入力し、集計などを行って分析すること。
- 表計算ソフトを利用してシミュレーションを行うこと。
- 集計やシミュレーションによって得られたデータをグラフに表現すること。

7.3 表計算ソフトの基本構造

7.3.1 セルの内容

表計算ソフトは「セル」が縦横に基盤目状の並んだ「シート」で構成されています。横方向の並びを「行」、縦方向の並びを「列」と呼びます。

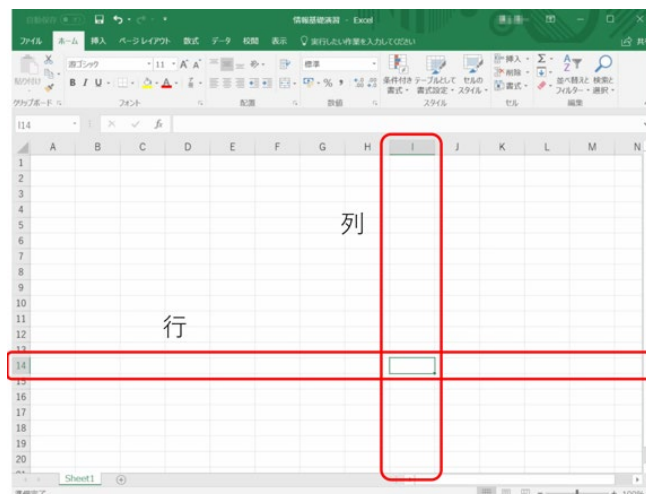


図 54 表計算ソフトのワークシート，行，列，セル

セルには「値」として数値や文字列が入力できるほか、他のセルの値を使った計算式を入力できます。図 54 の例ではセル A6 には数値ではなく式が入力されています。

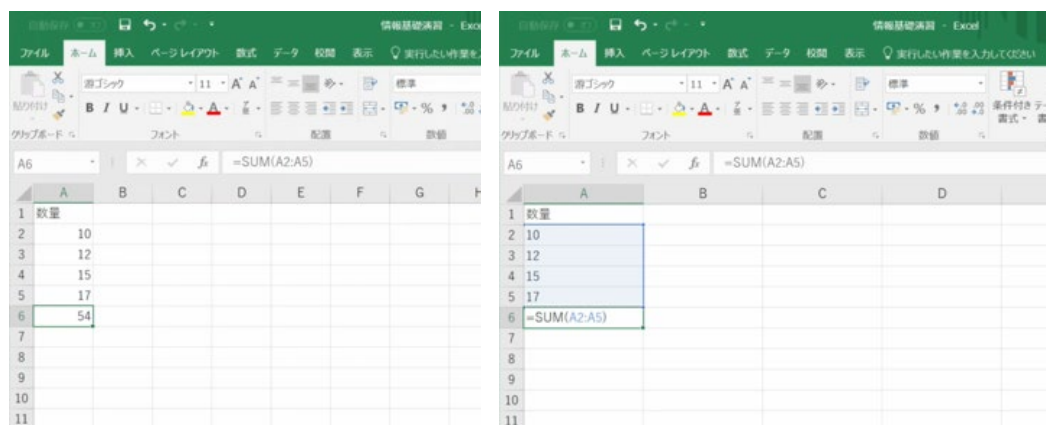


図 55 セルの内容

（右側の図は意図的に式が見える表示をしています。）

7.3.2 セルの式を表示する

通常、式を入れたセルでは計算結果が表示されます。式そのものを見えるようにするには以下の方法を取ります。

- 「ファイル」→「オプション」→「詳細設定」→「シートで作業する時の表示設定」の中で「計算結果の代わりに数式をセルに表示する」をチェック)
- また、計算結果を表示するか、式を表示するかの切り替えは Ctrl キーとシフトキーを押しながら 「`」 キーを押すことでもできます。

7.3.3 他のセルの値の参照

1) セルの値の参照

セルに式を書く際に他のセルの値は列の名前と行の番号を表記することで参照できます。例えば

=A2+A3+A4+A5

とすれば A 列の 2 行～5 行のセルの値の和を計算できます。

2) 矩形領域の指定

合計を計算する sum 関数などで矩形の領域の複数のセルを指定する際には例えば

=sum(A2:A5)

のように左上と右下のセルをコロン「:」でつないで表記します。

3) 相対参照と絶対参照

式を他のセルにコピーした場合、式の中で参照しているセルは式のあるセルからの相対的な位置を保存してコピーされます。例えば A6 のセルにある

=sum(A2:A5)

を隣の B6 にコピーするとセルの内容は

=sum(B2:B5)

となります。これを**相対参照**と言います。

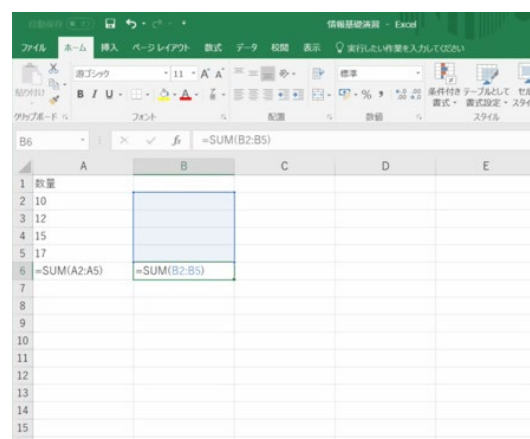


図 56 領域の参照

これに対して、コピー先に関わらず特定のセル（行や列）を参照したい場合には行や列の名前の前に \$ 記号をつけます。これを絶対参照と呼びます。下の例では B2 のセルに A2 のセルの 2 割増しを計算するため定数 1.2 をセル C2 に入れて、以下のように**絶対参照**で

=A2*\$C\$2

と表記しています。これを B3～B5 にコピーすると「A2」に相当する部分は相対参

照ですのでそれぞれ A3～A5 に変わりますが、C2 については同じセルを参照します。

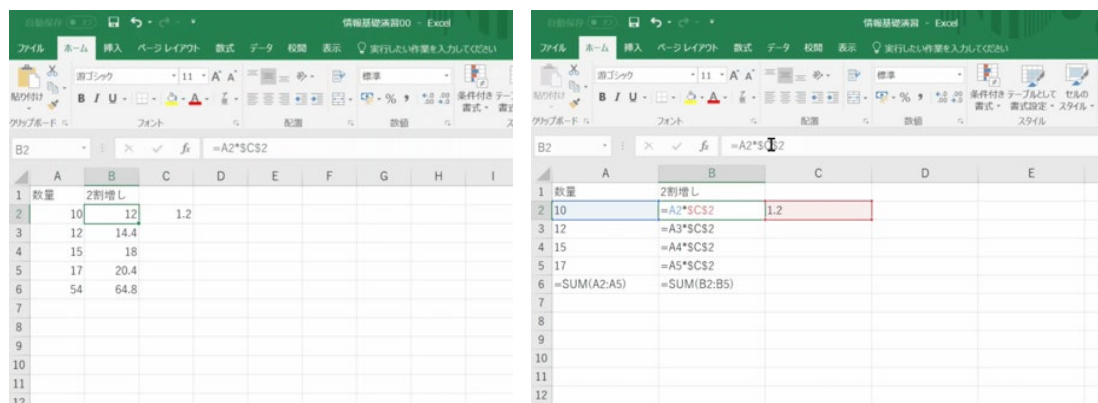


図 57 相対参照と絶対参照

- 表の特定のセルは「A6」のように列のアルファベットと行の番号の組み合わせで示される
- セルの値を参照し、演算記号や関数を使って計算ができる
- セルの参照には絶対参照と相対参照があり、絶対参照の際にはアルファベットや番号の前に\$マークを挿入する

図 58 表計算ソフトの基本構造

Great Idea! 表計算ソフトの開発

Dan Bricklin と Bob Frankston は 1979 年に表の中に数値だけでなく、表内の他のデータを参照する計算式を書けるようにし、自動的に計算するという表計算ソフト VisiCalc を Apple II コンピュータ用に開発しました。これは Apple II コンピュータのキラードアプリケーションソフトとなりました。現在の Excel などでも用いられている「A1」などのセルの参照表記は当時の VisiCalc の実装から継承されています。このアイデアは大量のデータの集計を誰もが行えるようにする画期的なものです。

演習 21. 相対参照と絶対参照

本節で示した例を実際に Excel で作成し、式の中で他のセルの値を参照することと、その際の相対参照、絶対参照の機能の違いを確認してください。

7.4 データの入力方法

7.4.1 データの収集と分析

ここでは調査などでデータを集め、それを表計算ソフトに入力し、分析することを考えます。その際、以下のような作業が必要になります。

- 調査の概要など必要な事項を記録すること。
- データを表計算ソフトに入力すること。その際、さらに以下の点に注意をします。
 - 後のデータ処理が容易な**表の構成**を用いること。
 - 誤りが極力少なくなるような**データの入力の工夫**をすること。
 - 入力されたデータが正しいかどうかを**確認**すること。

7.4.2 表の構成

1) シートの使い方

- 調査の概要なども Excel のシートに記載するようにします。Sheet1 を用いることにし、シートの名前を適宜変更します。
 - シート名を変更するには、変更したいシート名の上でダブルクリックします。または、そのシート名を右クリックし「名前の変更」を選択します。
- 調査で得られるデータ Sheet2 に記載することにします。同じくシートの名前を適宜変更します。
 - シートを増やすにはシート名の右にある「+」マークをクリックします。または、既存のシート名を右クリックし「挿入」を選択することでもシートを増やせます。
- 集計結果などは Sheet3 以降を使うようにします。
- シートを削除するには、削除するシート名上で右クリックし「削除」を選択します。
- シートを複製したり、他のファイルに移動させるには、シート名を右クリックし「移動またはコピー」を選択します。元のシートを残したい場合は表示された画面で「コピーを作成する」にチェックを入れます。

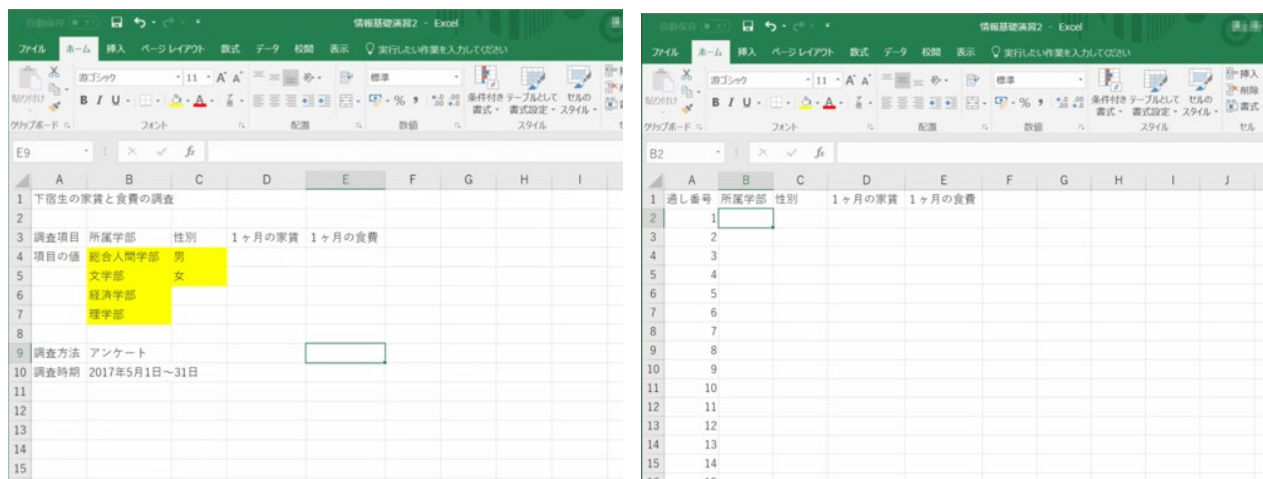


図 59 シートの使い方

2) データ表の構成

アンケート調査など、多くのデータは1件のデータが複数の項目で構成され、調査結果は多くの件数のデータで構成されます。これをシートに記載するには以下の方法を標準に行います。

- 1件のデータを1行で構成します。
- 表の1行目には**項目の見出し**を書きます。
- 1列目は通し番号用に使うと便利です。
- データは2行目以降に入力して行きます。

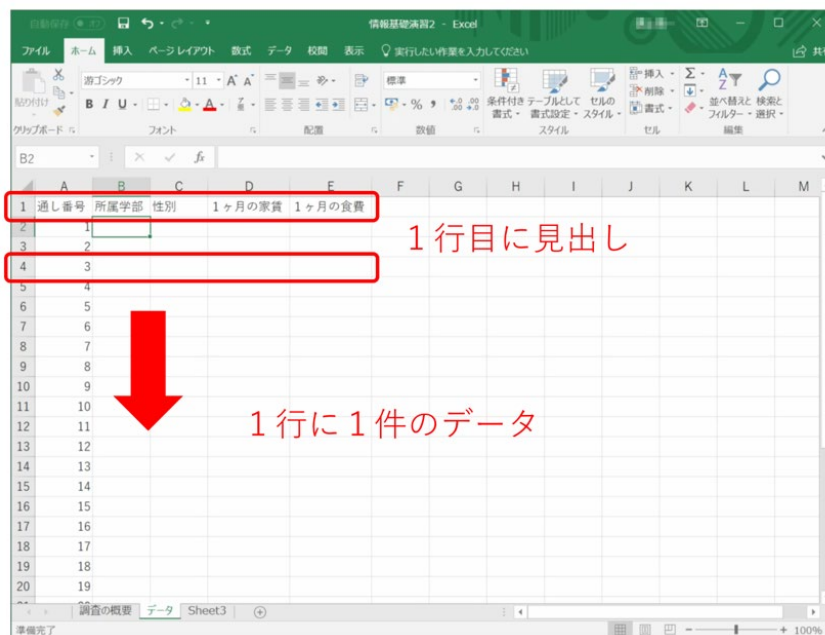


図 60 表の構成の基本

連続する数字を入力するには、「通し番号」の A2 に「1」を入力し、Ctrl キーを押しながら選択範囲の右下角を下側にドラッグします。または、「ホーム」リボン上で「フィル」から「連続データの作成」を選ぶことでも連続した数字を入力できます。後者の場合、連番を振る方向（行または列）や振りたい番号の最大値（停止値）を指定することもできます。

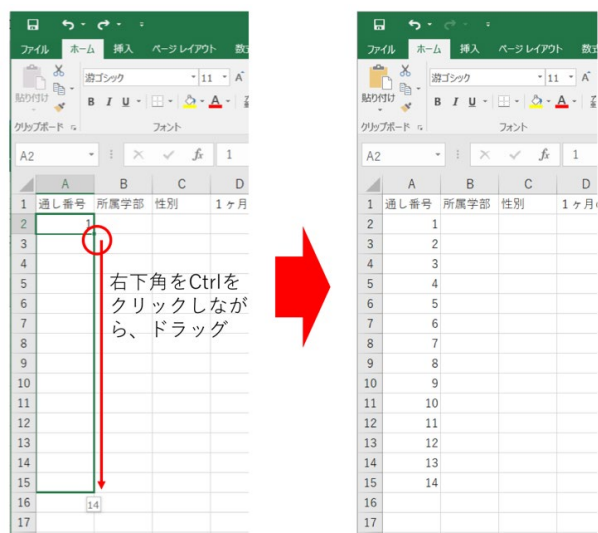


図 61 連続する数値の入力

7.4.3 入力の制限

データの入力はたいへん労力を要し、なおかつ人が行うため**必ず誤りが発生する**と考えることが重要です。このため作業効率を高める工夫と誤りを極力なくする作業手順が重要です。

1) ウィンドウ枠の固定

見出しや通し番号を常に見えるようにしておけばデータの入力や確認が容易になります。このために「ウィンドウ枠の固定」という操作を行います。見出しや通し番号以外のデータの左上のセル（セル B2）を選んだ状態で、「表示」タブのリボンから「ウィンドウ枠の固定」を選びます。

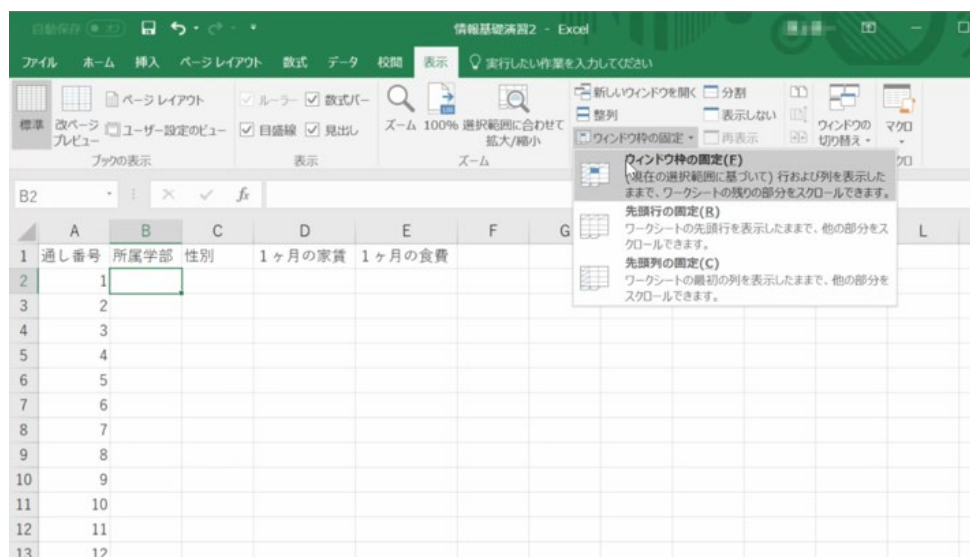


図 62 ウィンドウ枠の固定

2) データ入力規則の設定

データの内容が例えば「学部」とか「性別」などのように**特定の文字列**に限られている場合、これを**直接タイプ入力してもらおうと、表記が揺らぐ**などして多くの誤りが発生します。これを予め定められたものから選べるようにするには以下のようにデータの入力規則を設定します。

- 入力候補の文字列を例えば調査の概要のシート(Sheet1)に作成します。
- データ用のシートで、入力対象となるセルを選びます。
- データタブを選んで、リボンから「データの入力規則」を選び、「設定」として「リスト」を選びます。
- 「元の値」として、先に設定した候補の文字列を選びます。調査の概要のシートに移動し、候補文字列の先頭を選び、Shift キーを押したまま、最後の候補文字列を選べば、候補文字列を設定できます。
- 入力規則を設定したセルを下方に必要な分だけコピーします。

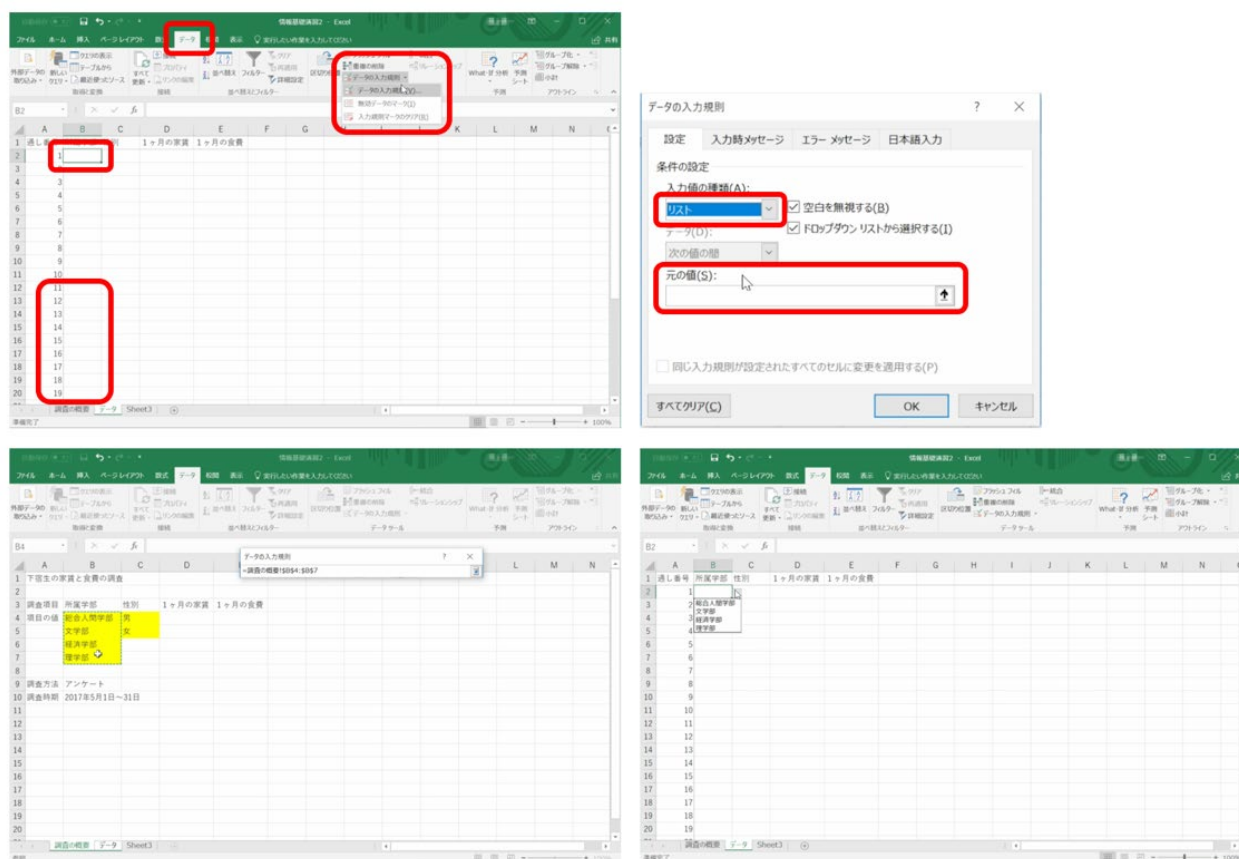


図 63 入力の制限方法

2 人で作業できる場合などは、

- 1 人が読み上げと入力の確認、
- もう 1 人が表計算ソフトの操作を担当する

と効果的です。

データの入力が終わったら、必ず元のデータが正しく入力されているか照合するようにします。

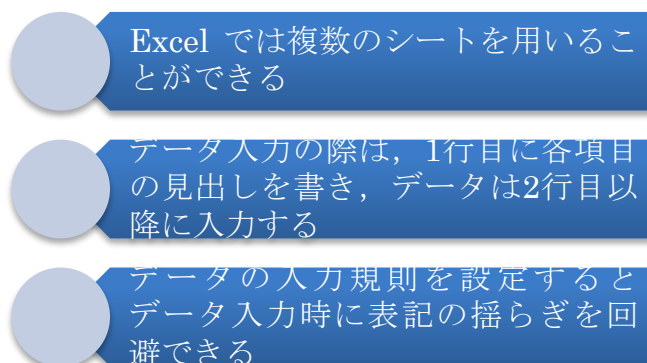


図 64 データの入力方法のまとめ

演習 22. データの作成

表 6 に示したデータを本節で示した手順に沿ってシートとして構成し、データ入力規則の設定などして入力してください。ただし学部は総合人間学部、文学部、経済学部、理学部が現れます。

表 6 演習用のサンプルデータ（架空のものです）

通し番号	学部	性別	1ヶ月の家賃	1ヶ月の食費
1	文学部	女	41000	16000
2	文学部	女	39000	13000
3	経済学部	男	22000	15000
4	経済学部	男	48000	14000
5	経済学部	男	26000	15000
6	文学部	女	51000	19000
7	文学部	女	49000	21000
8	総合人間学部	男	28000	19000
9	文学部	女	40000	11000
10	理学部	女	45000	10000
11	理学部	男	37000	11000
12	総合人間学部	女	32000	14000
13	理学部	女	32000	10000
14	経済学部	男	30000	12000
15	理学部	男	46000	17000
16	経済学部	男	29000	13000
17	総合人間学部	女	51000	27000
18	経済学部	男	34000	14000
19	文学部	女	23000	14000
20	経済学部	男	20000	14000
21	文学部	男	45000	17000
22	文学部	男	21000	16000
23	経済学部	女	50000	14000
24	理学部	男	48000	25000
25	経済学部	女	51000	26000
26	文学部	女	24000	23000
27	経済学部	男	24000	24000
28	文学部	女	26000	22000
29	経済学部	男	32000	24000
30	文学部	女	45000	23000

7.5 データの並べ替え

データを使った分析では特定の項目に注目して、その値でデータを並べ替えたいことが少なくありません。

- データの並べ替えのことは「整列」や「ソート(sort)」とも呼ばれます。
- データの値が小さいものから大きくなる順に並べることを「昇順」と言います。
- 逆に大きいものから小さくなる順に並べることを「降順」と言います。
- 文字列は文字を表すコードの順で並べ替えられます。漢字の場合は読みの項目などを作ってそれで並べ替える必要があります。

前節の演習 21 で作ったデータを並べ替えてみましょう。

- データのあるセルをどれか選びます。
- データタブを選び、リボンの中の「並べ替え」を選びます。
- 最優先キーを並べ替えたい項目に、順序を昇順や降順に選びます。
- 先頭行をデータの見出しとして使用する、が指定されていることを確認します。
- OK を選ぶと並べ替えが完了します。

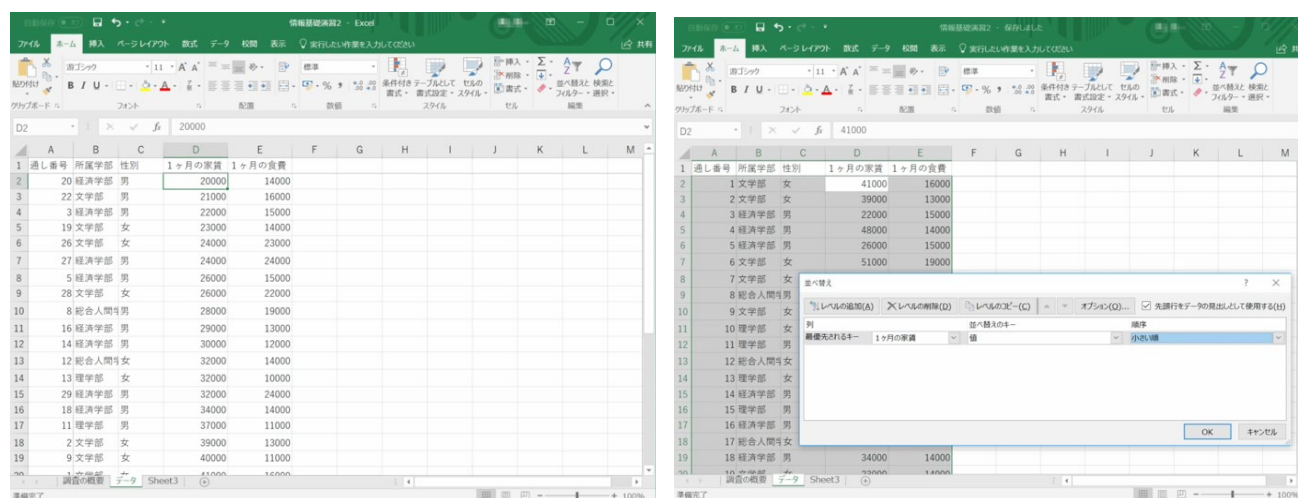


図 65 データの並べ替え

演習 23. データの並べ替え

演習 21 で入力したデータについて家賃や食費で並べ替えを行い、それぞれ高いものから 3 件、低いものから 3 件の値を書きだしてください。

家賃の高いもの 3 件			
家賃の低いもの 3 件			
食費の高いもの 3 件			
食費の低いもの 3 件			

7.6 データの集計

データを集計するために Excel にはいくつかの関数が用意されています。

7.6.1 個数を数える

表 7 個数を数える関数

関数名	意味	例
count	範囲内の数値の個数を数える	count(A2:A31)
counta	範囲内の空白でないセルの数を数える	counta(A2:A31)
countif	範囲内で条件を満たすセルの個数を数える	countif(B2:B31,"工学部") countif(D2:D31,">20000")
countifs	範囲内で複数の条件を満たすセルの個数を数える	countifs(B2:B31,"工学部", D2:D31,">20000")

countifs 関数では異なる列の範囲に対して条件を設定して数えることができます。例えば上の表に示した例では

- B 列の 2 ～31 行の値が“工学部”で、
 - D 列の 2～31 行の値が 20000 より大きい、
- 行の数を数えます。

7.6.2 合計を求める

表 8 合計を求める関数

関数名	意味	例
sum	範囲内の数値の合計を求める	sum(D2:D31)
sumif	範囲内の条件を満たすセルの値の合計を求める	sumif(D2:D31,">20000") sumif(B2:B31,"工学部",D2:D31)
sumifs	範囲内で複数の条件を満たすセルの値の合計を求める	sumifs(D2:D31,B2:B31,"工学部", C2:C31,"男")

sumif の 2 つ目の例は条件検索をする範囲が B 列 (B2:B31) ですが、実際に合計するのは D 列という場合の書き方です。

sumifs 関数は複数の条件を書けますが、最初に合計する範囲指定して、後で条件を検索する範囲と検索条件を対にして必要な数だけ書きます。

7.6.3 平均を求める

平均を求める関数には average, averegeif, averageifs などがあります。使い方は sum, sumif, sumifs 同様で、合計の代わりに平均が求まります。

演習 24. データの集計

演習 21 で作成したデータについて、sheet3 に“集計”と名前を設定して、以下を求めなさい。

- 全体のデータ件数，家賃，食費の総和と平均
- 学部別のデータ件数，家賃，食費の総和と平均
- 性別のデータ件数，家賃，食費の総和と平均

別のシートの範囲を選択するにはシート名に ! を付けて、例えば「データ!B2:B31」と書きます。

	A	B	C	D	E	F	G	H	I	J
1	学部名	件数	家賃総和	家賃平均	食費総和	食費平均				
2	総合人間学部	3	111000	37000	60000	20000				
3	文学部	11	404000	36727.27	195000	17727.27				
4	経済学部	11	366000	33272.73	185000	16818.18				
5	理学部	5	208000	41600	73000	14600				
6	性別	件数								
7	男	15	490000	32666.67	250000	16666.67				
8	女	15	599000	39933.33	263000	17533.33				
9	全体	30	1089000	36300	513000	17100				
10										
11										
12										
13										
14										
15										

図 66 データの集計

また、検索条件は直接書く代わりに、検索条件を書いたセルを参照してもかまいません。上の例でセル C2 は条件として A2 を参照する形で以下のように書かれています。

=SUMIF(データ!\$B\$2:\$B\$31,A2,データ!\$D\$2:\$D\$31)

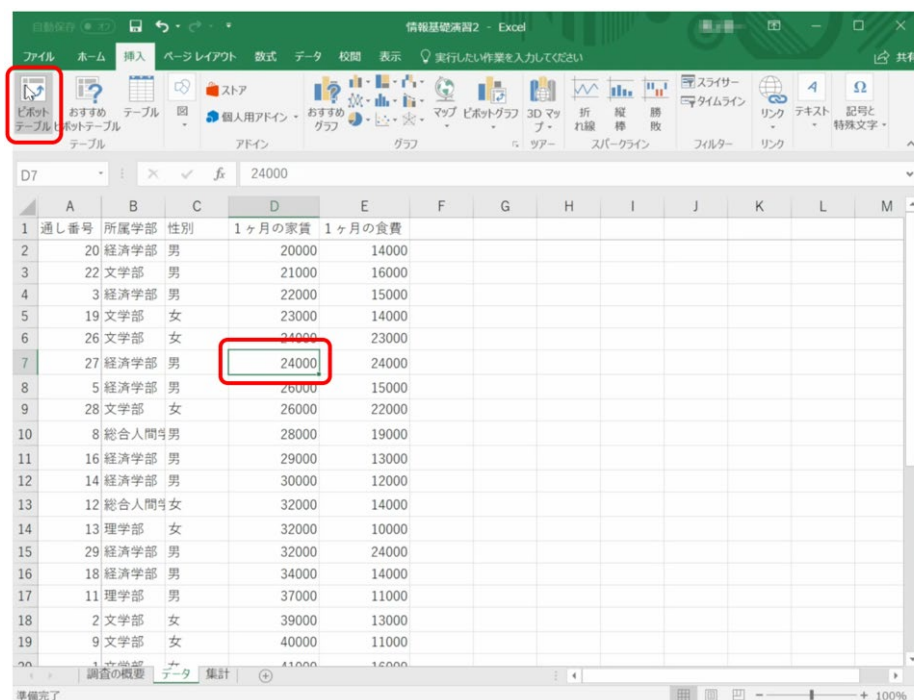
また、データ範囲の設定は絶対参照で書かれていますが、これは下方にコピーした際に検索条件は A3, A4, A5 を参照しつつ、データ範囲は変えないためです。

7.7 データのクロス集計

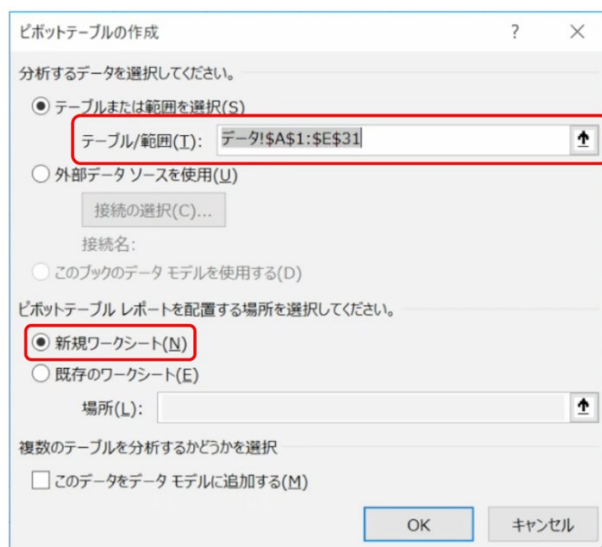
演習 21 で作成したデータに対して「学部別」と「性別」両方の条件ごとにデータ件数を求めたり、総和や平均を求めたりすることをクロス集計といいます。Excel でクロス集計するためには「ピボットテーブル」を使います。

- データのあるシート上でデータのあるセルを選びます。

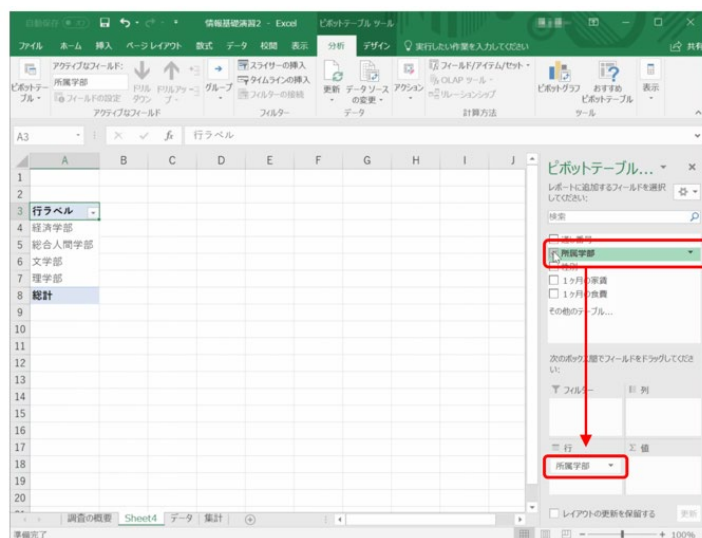
- 挿入タブで表示されるリボンから「ピボットテーブル」を選びます。



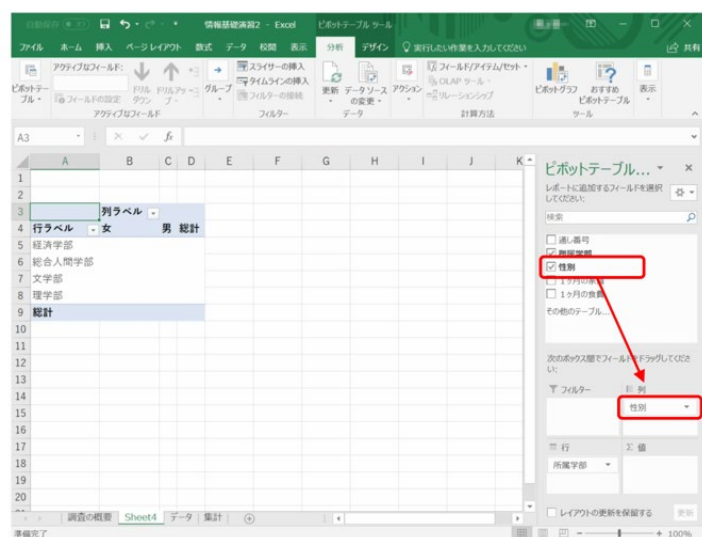
- データ範囲と新規ワークシート上に作成を確認して「OK」を押します。



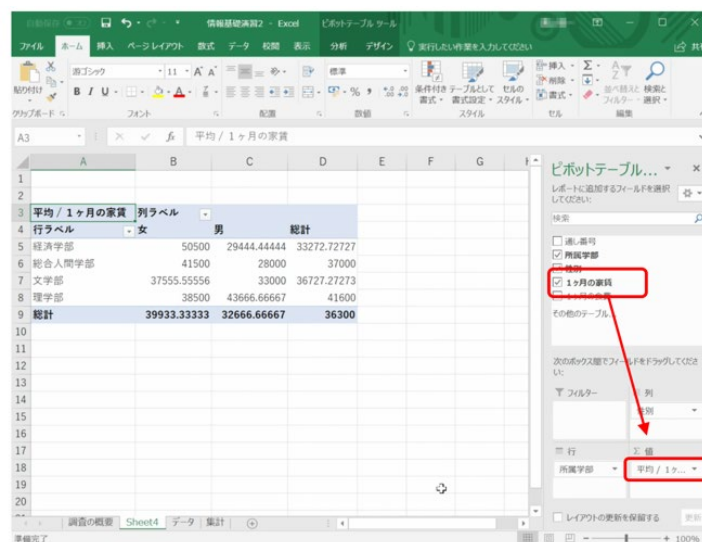
- ピボットテーブルのフィールドリストから所属学部を選び、行ラベルにドロップします。



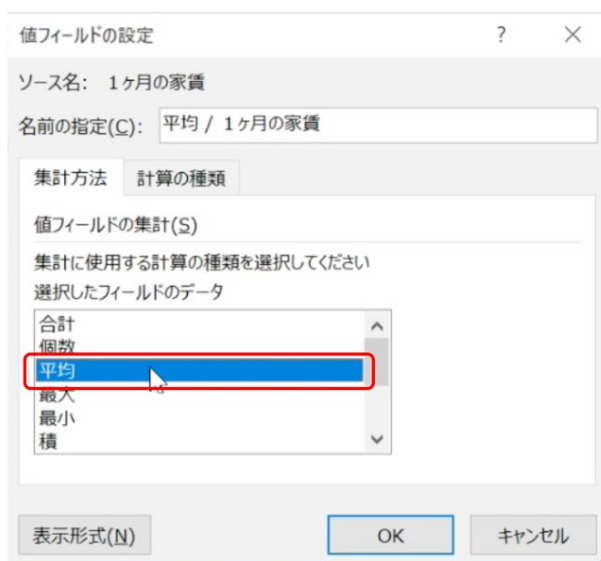
- 性別を選び、列ラベルにドロップします。



- 1ヶ月の家賃を選び、Σ値にドロップします



- Σ 値の合計 / 1 ヶ月の家賃をクリックし、合計ではなく平均を選びます。



- 作られたシートのピボットテーブル以外のセルをクリックすればフィールドリストは消えます。ピボットテーブル上のセルを選べば再表示されます。

演習 25. クロス集計

演習 21 のデータに対してピボットテーブルを用いて学部別、男女別のデータ件数を求めなさい。

		性別 / 1ヶ月の家賃	
		男	女
学部	1ヶ月の家賃	男	女
経済学部	2	9	11
総合人間学部	2	1	3
文学部	9	2	11
理学部	2	3	5
総計		15	30

8. 表計算ソフトによるデータ処理（2）

8.1 学習の目的

- 表計算ソフトを用いて行うシミュレーションについて事例を用いて学びます。

8.2 シミュレーションとは

シミュレーション (simulation) とは考察の対象をモデルを用いて模擬することです。シミュレーションの対象としては物理的なシステムであったり、人が含まれる社会的なシステムであったりします。また、対象の変化の取扱いによって

- 定常的な状態にある様子を模擬する静的な状況のシミュレーション
- 時間とともに変化する様子を模擬する動的な状況のシミュレーション
- ランダムな要因を含むシミュレーション

という見方もできます。ここでは「**動的な状況のシミュレーション**」を取り上げて表計算ソフトでシミュレーションを実施することを学びます。

8.3 動的な状況のシミュレーションの構成要素

動的な状況を表すモデルの構成要素は以下のものです。

- 時間と共に変化する量 (変数)、これが計算の対象となります。
- 変数の変化を規定するルール。例えば1時刻前の値から次の時刻の値を得る関数 (式) など。
- 変数の最初の時刻での値 (初期値)
- パラメータ (parameter)。値の変化を規定するルールの中で参照される値をパラメータと言います。それ自体は時間とともに変化するものではありませんが、パラメータを変えたときに変数の値の変化の仕方などを考察します。

8.4 借入金の返済のシミュレーション

借入金には元本に利息をつけて返済しなければなりません。借入金と利息の一部が

つを返済して行くことで借入残高がどのように変化するかをシミュレーションしてみましよう。

- 借入金 は 1,000 万円とします。
- 利息は年に 3% とします。ただし端数は切り捨てるものとします。
- 1 年当たりの返済額は a 円とします。
- 借り入れは年初めに行うものとし、その年の終わりに利息がつくものとします。これから返済額を差し引いたものが、翌年の借入金残高となるものとします。すなわち

$$\text{翌年の借入金残高} = \text{int}(\text{前年の借入金} \times (1 + 0.03) - a)$$

ここで $\text{int}()$ は切り捨てて整数にする関数を意味します。

これを Excel で計算し、グラフ表示したものが図 66 です。上の図は計算結果を、下の図は式を表示したものです。計算結果を表示するか、式を表示するかの切り替えは Ctrl キーとシフトキーを押しながら 「`」 キーを押します。

- 式以外の箇所は文字列や値を直接入力します。E1, E2 セルにはそれぞれパラメータとして利率と 1 年当たりの返済額を設定します。
- 式はまず以下のように入力します

B3 セル $\text{=MAX(INT(B2*(1+\$E\$1))-\$E\$2),0)}$

C2 セル $\text{=MIN(\$E\$2,INT(B2*(1+\$E\$1)))}$

それぞれ E1 セルの利率, E2 セルの毎年の返済額は絶対参照するために \$E\$1, \$E\$2 と書いています。MAX や MIN を用いているのは借入金残高が 0 以下にならないため、そして毎年の返済額が返済すべき額より大きくなったときに適切な値になるようにするためです。

- これらのセルをそれぞれ下方にコピーすることで表を完成させます。
- グラフの作成については次章を参考にしてください。

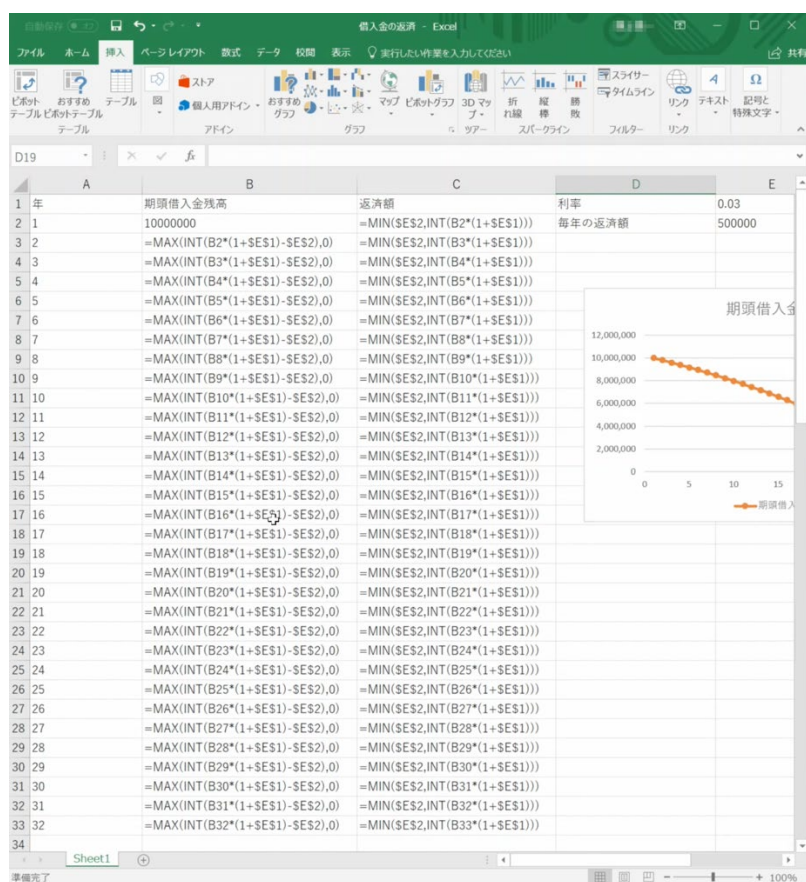
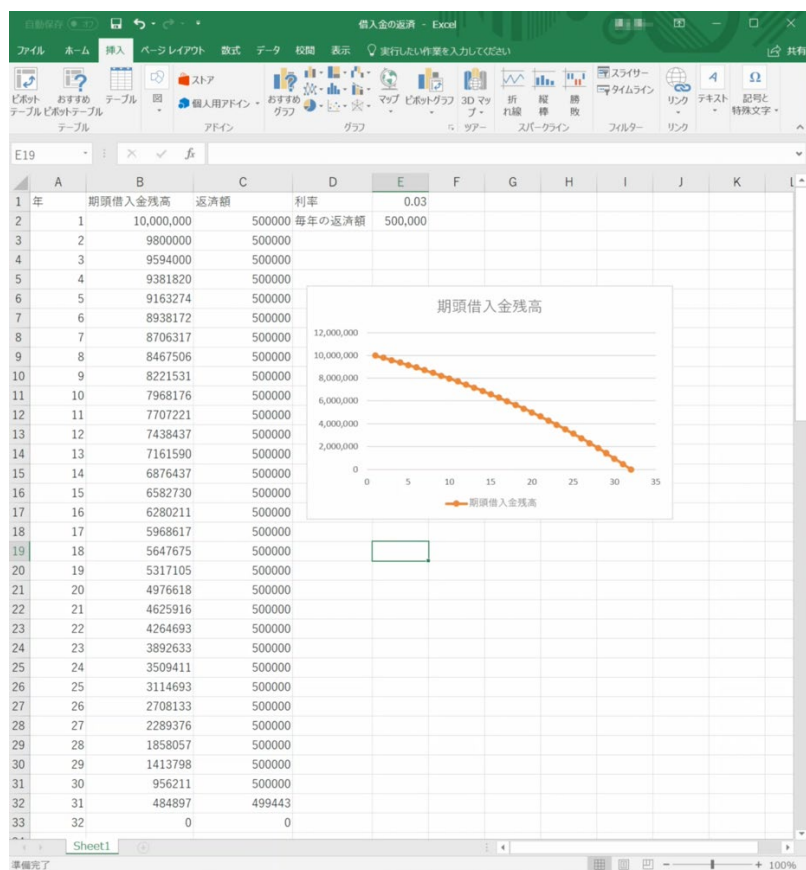


図 67 借入金返済のシミュレーション。下の図はセルの中の式を表示したもの

演習 26. 返済額を変えたシミュレーション

返済額を変えてシミュレーションを行い、借入金の残高の変化について考察しなさい。

8.5 バネ・ダンパモデルのシミュレーション

ここでは機械的なシステムのシミュレーションとしてバネ・ダンパモデルを取り上げます。

8.5.1 モデル

バネにおもりを吊るすことで、バネの伸びからおもりの重さを計測するのが「ばねばかり」の原理ですが、バネ内でのエネルギーの損失が少ない場合にはおもりを吊るすと振動してしまい、バネの張力とおもりの重力が釣り合う長さでなかなか止まりません。このため動きを制動するための抵抗となる「ダンパ」が用いられます。このようなバネとダンパ、おもりから構成される仕掛けでのおもりの運動をシミュレーションすることを考えます。

おもりの質量を m , バネ定数を k , 重力加速度を g , ダンパの係数を c , おもりの位置 (バネに力がかかっていない状態の位置からのずれ) と速度を鉛直方向下向きを正にとってそれぞれ x, v とし, x はバネの自然な長さを原点にとるとするとおもりの運動方程式は次の式であたえられます。(1)式は速度の定義そのものです, (2)式は左辺の加速度 (速度の微分) が右辺のように定数である重力, 位置に比例するバネの復元力と速度に比例するダンパの抵抗力で与えられることを示したものです。

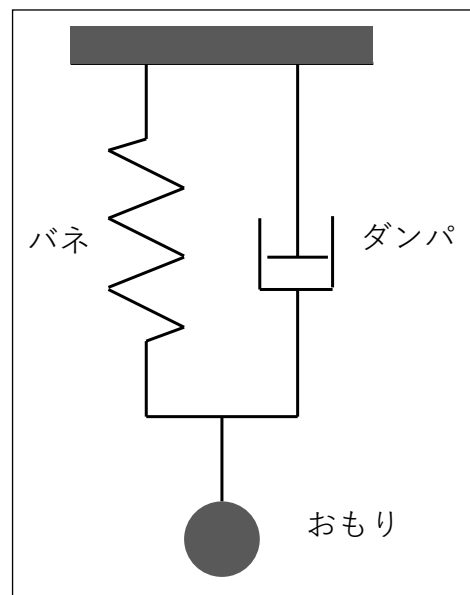


図 68 バネ・ダンパモデル

$$\frac{dx(t)}{dt} = v(t) \quad (1)$$

$$\frac{dv(t)}{dt} = g - \frac{k}{m}x(t) - \frac{c}{m}v(t) \quad (2)$$

またおもりの初期値は $x(0) = 0$, $v(0) = 0$ とします。

8.5.2 微分方程式のオイラー近似

微分方程式は連続時間の関数についての方程式ですから、そのままでは数値計算できません。ここでは短い時間間隔 Δt でおもりの位置や速度を計算して行くものとしします。すなわち $x_0 = x(0)$, $x_1 = x(\Delta t)$, $x_2 = x(2\Delta t)$, ..., $v_0 = v(0)$, $v_1 = v(\Delta t)$, $v_2 = v(2\Delta t)$, ... を考えます。そして、時刻 $n \Delta t$ から $(n+1) \Delta t$ の間は式(1), (2) で与えられる微係数は時刻 $n \Delta t$ の値で一定であると近似します。これにより、以下の差分方程式が得られ、数値計算ができます。

$$x_{n+1} = x_n + \Delta t \quad v_n \quad (3)$$

$$v_{n+1} = v_n + \Delta t \left(g - \frac{k}{m}x_n - \frac{c}{m}v_n \right) \quad (4)$$

このような近似はオイラー近似と呼ばれ、微分方程式を数値計算する方法としては最も簡単なものです。ただし、時間間隔 Δt を大きく取ると精度が低下します。一方で小さく取ると数値計算に必要な計算量が大きくなります。本格的な数値計算にはより精度の高いさまざまな計算方法が開発されています。

8.5.3 表計算ソフトでのシミュレーション

式(3), (4)に基づいて Excel で行ったシミュレーションを図 68 に示します。パラメータは次の表のように設定しています。

表 9 バネ・ダンパモデルのパラメータのセルへの割り付け

パラメータ	セル
おもりの質量 m	E2
バネ定数 k	E3
重力加速度 g	E4
ダンパの係数 c	E5

時間の刻み Δt

E6

これを用いて時刻，位置，速度の更新は例えば 3 列目 (A3, B3, C3) では以下のよう
に設定しています。借入金の返済シミュレーションと同様にパラメータはコピー
した際に参照位置がずれないように絶対参照を行っています。

$$A3 = A2 + \$E\$6$$

$$B3 = B2 + \$E\$6 * C2$$

$$C3 = C2 + \$E\$6 * (\$E\$4 - \$E\$3 * B2 / \$E\$2 - \$E\$5 * C2 / \$E\$2)$$

これらのセルを下方にコピーすることで時刻，位置，速度を更新するシートを作成
できます。

演習 27. ダンパの係数についての分析

ダンパの係数を 0 (この場合，単振動を生じます。) から徐々に大きくするとおも
りの運動はどのように変化をするかシミュレーションを用いて考察しなさい。

演習 28. ダンパの係数が 0 のときの吟味

ダンパの係数が 0 の場合は理論的には単振動を生じるはずですが，数値計算では振
動の振幅がどのように変化するのか観察しなさい。

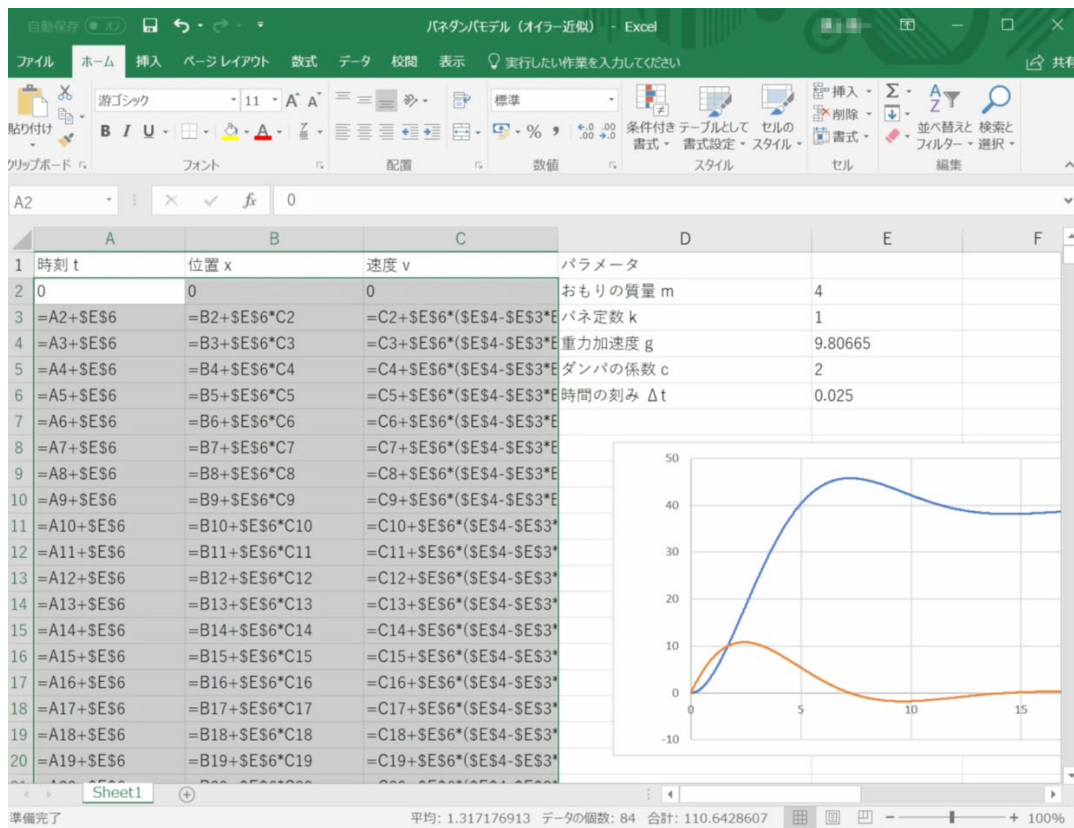
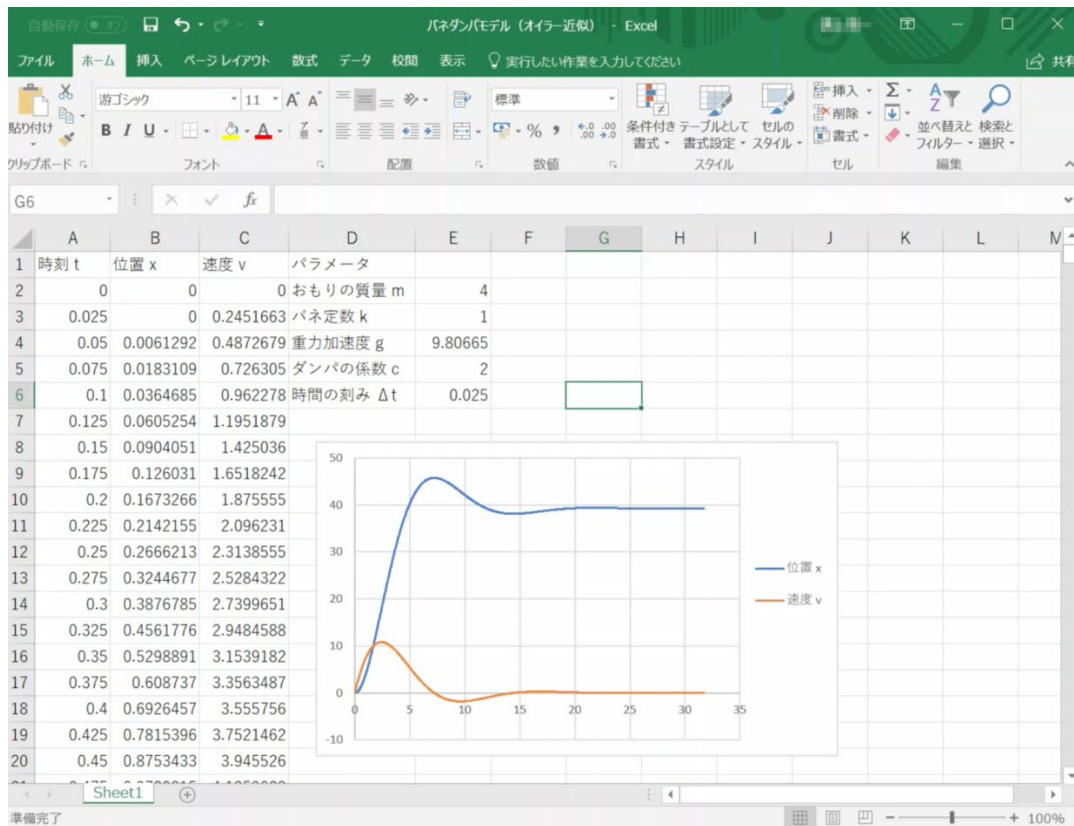


図 69 バネ・ダンパモデルのシミュレーション

8.5.4 静的な状況とその扱い

静的な状況をバネ・ダンパモデルで考えてみましょう。時間的に変化しないのですから、式(1), (2)で表される微係数が 0 になると考えます。すなわち

$$0 = v(t) \quad (5)$$

$$0 = g - \frac{k}{m}x(t) \quad (6)$$

これはバネの張力とおもりに働く重力がつりあっている状態にほかなりません。この例では容易に解くことができますが、より複雑な微分方程式で表された対象が静止している状態は微係数を 0 とおいて得られる連立代数方程式を解くことが求められます。

9. 表計算ソフトによるデータ処理（3）

9.1 学習の目的

- データをグラフとして表現することの意義と注意事項を考えます。
- Excel でグラフを作成する際の用語を学びます。
- Excel で散布図を描く方法を学びます。
- Excel で折れ線グラフや棒グラフを描く方法を学びます。
- Excel でヒストグラムを描く方法を学びます。

9.2 データをグラフで表示することの意味

演習 29. レポートなどではしばしばデータをグラフで表示します。なぜグラフで表示するのでしょうか？

演習 30. データをグラフで表示することは知的生産活動の中でどのようなことに役立つでしょうか。

9.3 データをグラフで表現する時の注意事項

次のグラフは演習 21 で示したデータについて横軸に 1 ヶ月の家賃を、縦軸に 1 ヶ月の食費をとってグラフに示したものです。レポートなどを行う際にこのようなグラフで示された場合に読み手に情報は十分に伝わるでしょうか。

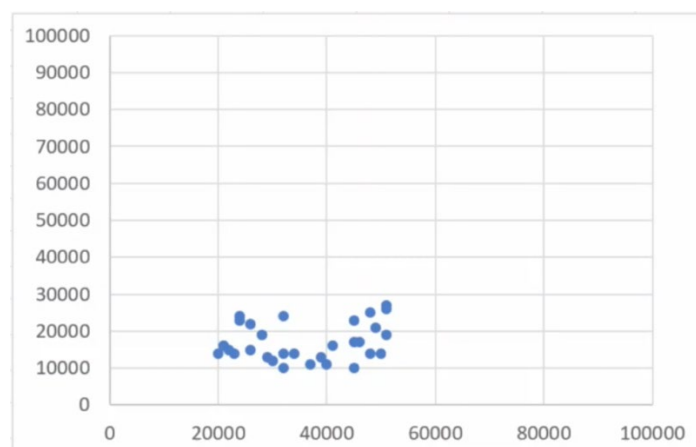


図 70 あまり適切でないプロット

演習 31. この問題点を挙げてみてください。

9.4 グラフを Excel で作成する際の用語

Excel でグラフを描く際にはグラフを構成する要素の名称を確認しておくとしかりやすいです。図 70 は言葉が示す対象を分かりやすくするため背景を着色しています。

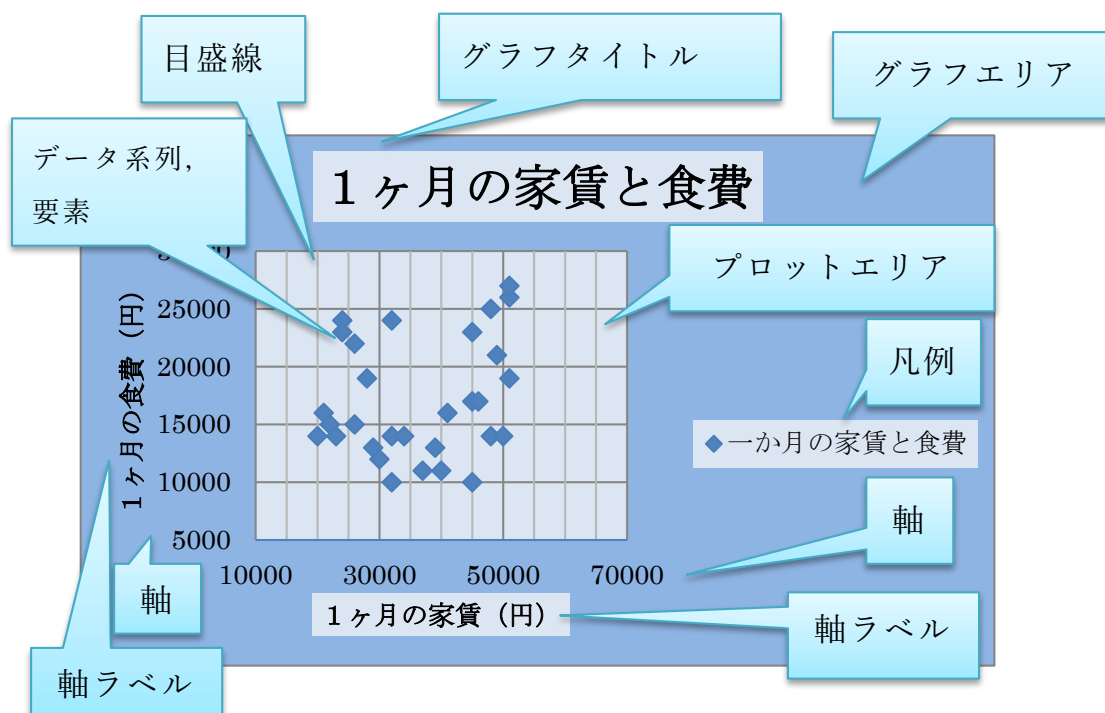


図 71 Excel でのグラフ各部の名称

9.5 散布図を Excel で描く

9.5.1 散布図の用途

先の例に示したグラフは横軸と縦軸の値を与えてデータをプロットするもので Excel では「散布図」という形式です。以下のような使い方があるでしょう。

- アンケート調査などのデータで2つの属性の関係を見るときに使います。
- 実験などで横軸の値を設定して、縦軸の値を計測し、両者の関係をグラフに表す。通常、折れ線グラフなどで描きますが、横軸の値を自由に与えてプロットする際には散布図を選びます。

9.5.2 散布図の作成手順

前回の演習 21 のデータを例に散布図を描く手順を以下に示します。

1. 散布図を描くデータ列を2つ（例では「1ヶ月の家賃」と「1ヶ月の食費」）を選びます。

情報基礎演習2 - Excel

ファイルホーム挿入ページレイアウト表形式データ視覚化表示

実行したい作業を入力してください

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

リボンメニュー

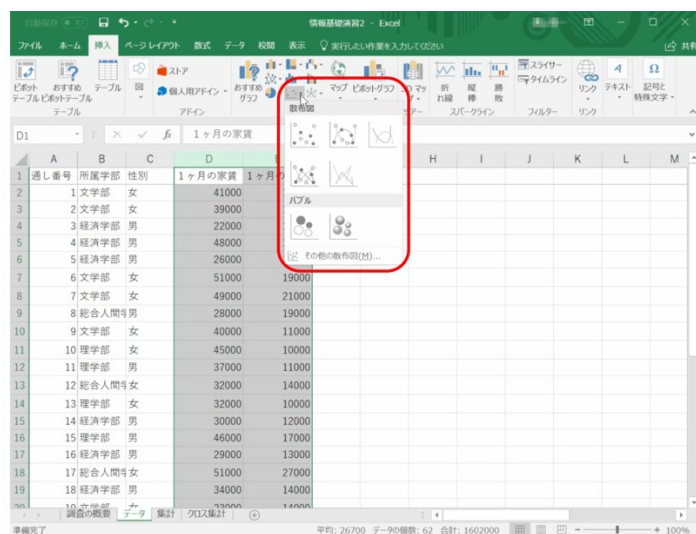
リボンメニュー

リボンメニュー

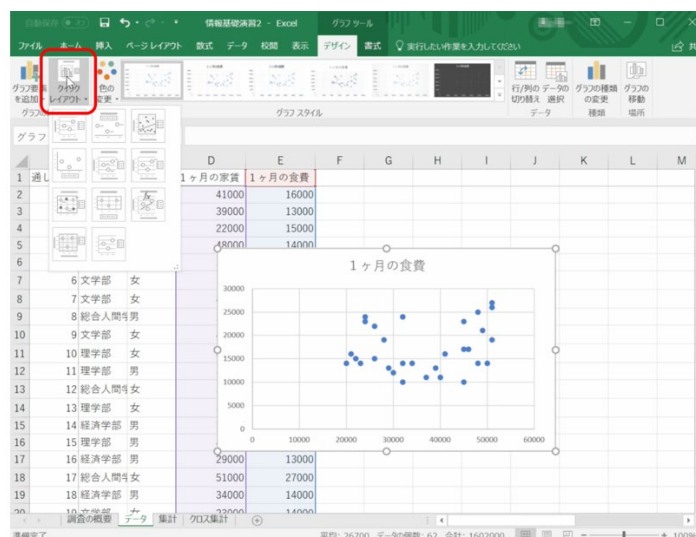
リボンメニュー

リ

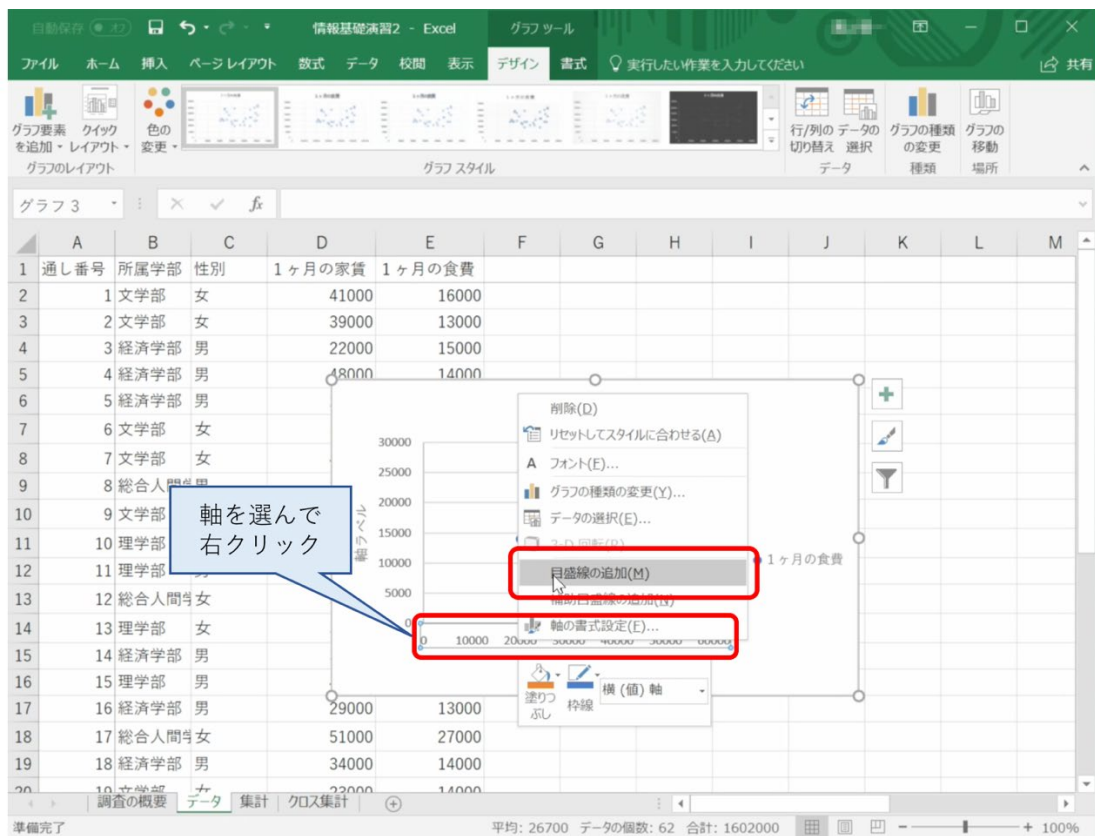
2. 「挿入」タブを選び、グラフのリボンの「散布図」を選び、プロットする形式（この場合はマーカーだけ）を選びます。



3. 描かれたグラフを選んで、リボンから適当なレイアウトを選びます。

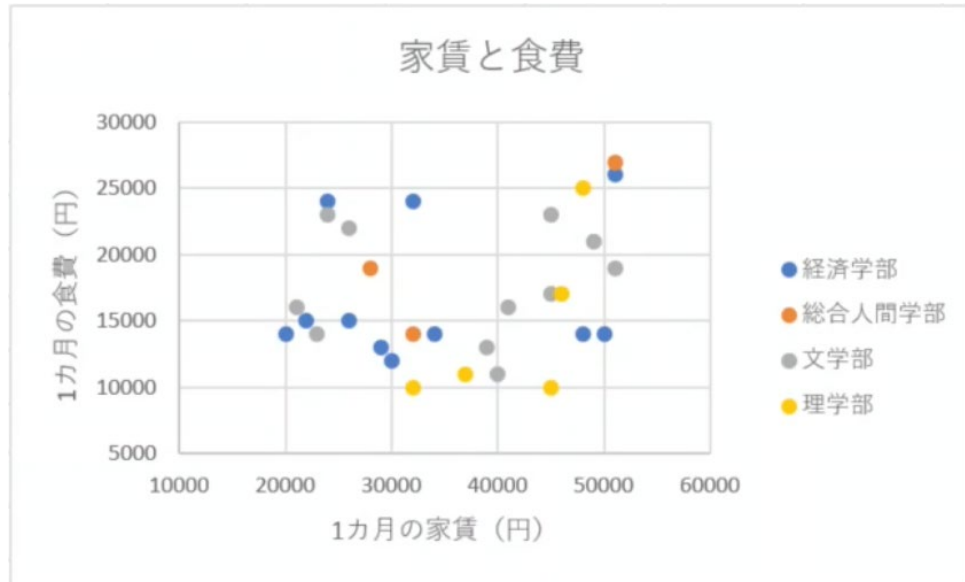


4. 横軸を選んで右クリックし、「目盛線を追加」を選びます。
 5. グラフタイトル，軸ラベル，凡例などを適宜修正します。

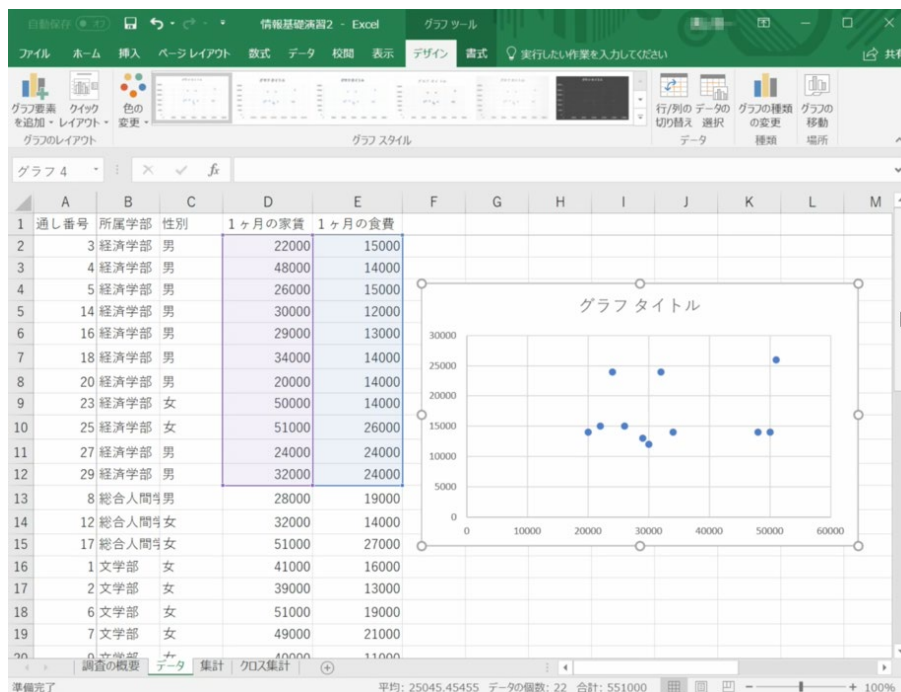


9.5.3 複数系列の散布図

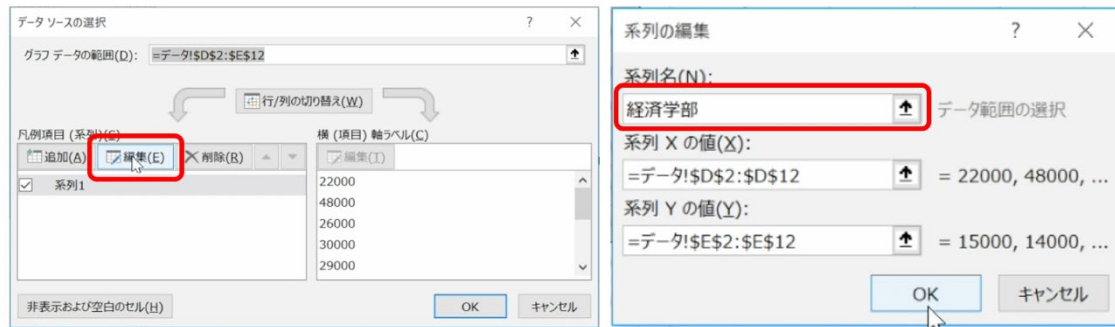
以下のような学部別の散布図を作成してください。



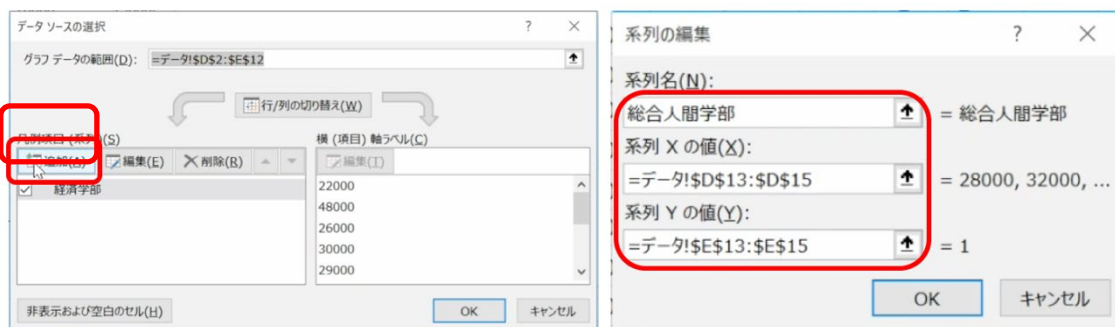
1. まず、データを学部で並べ替えます。
2. 経済学部について、家賃と食費のデータを選択して散布図を描きます。



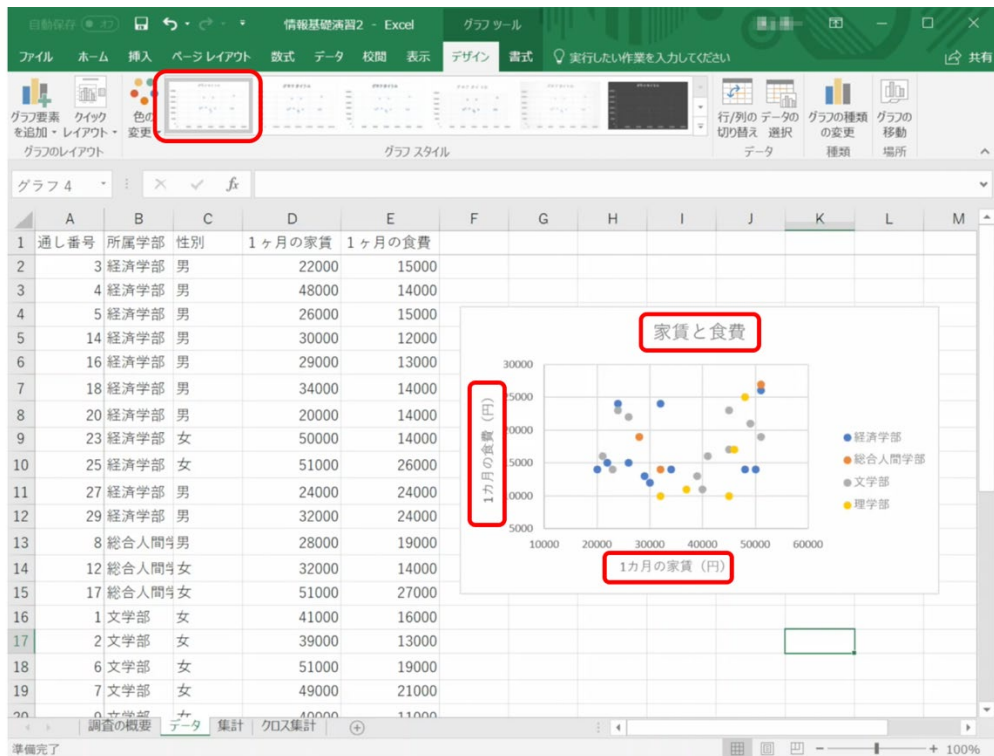
3. プロットされたデータを選んで右クリックし、データの選択を選びます。系列1を編集して系列名を設定します。



4. データの追加を選び、総合人間学部の家賃と食費のデータを設定します。



5. 同様に文学部と理学部についても作業します。
6. レイアウトを選びグラフタイトルや軸ラベル、目盛線などを設定します。



9.6 棒グラフや折れ線グラフを Excel で描く

Excel で棒グラフや折れ線グラフを描くことは散布図より分かりやすいと思います。次の例はデータを家賃の昇順で並べ替えて折れ線グラフでプロットしたものです。

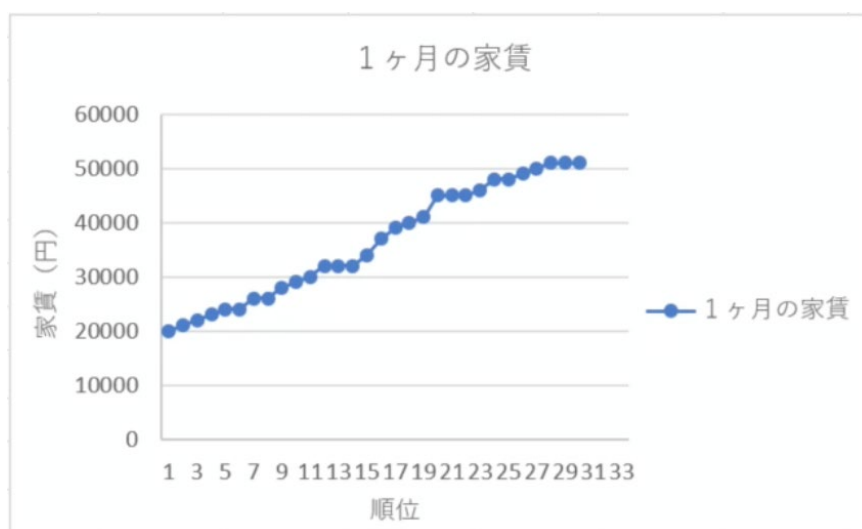


図 72 折れ線グラフの例

また、次のグラフはピボットテーブルを使ってクロス集計した家賃の平均を縦棒グラフで表示したものです。ピボットテーブル内で適当にデータを選択し、「挿入」→「縦棒」グラフ→「集合縦棒」と選ぶと自動的にピボットテーブルの内容をグラフ表示できます。

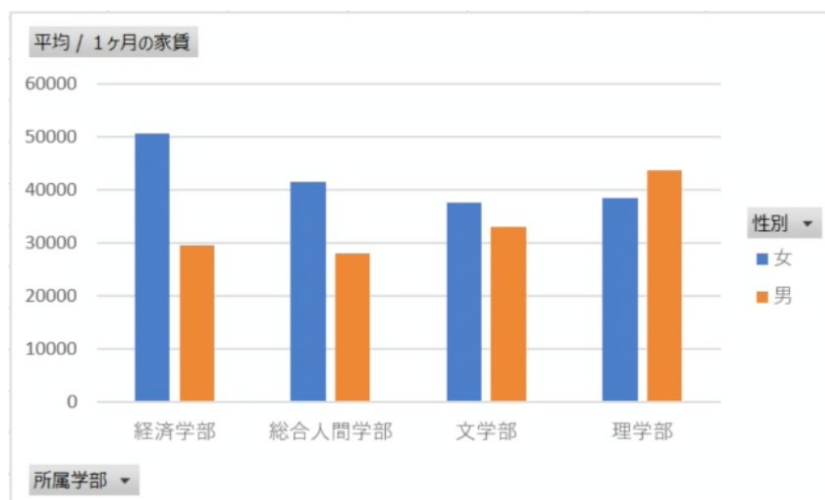


図 73 棒グラフの例

9.7 ヒストグラムを描く

以下のような家賃の分布をヒストグラムとして描いてみましょう。

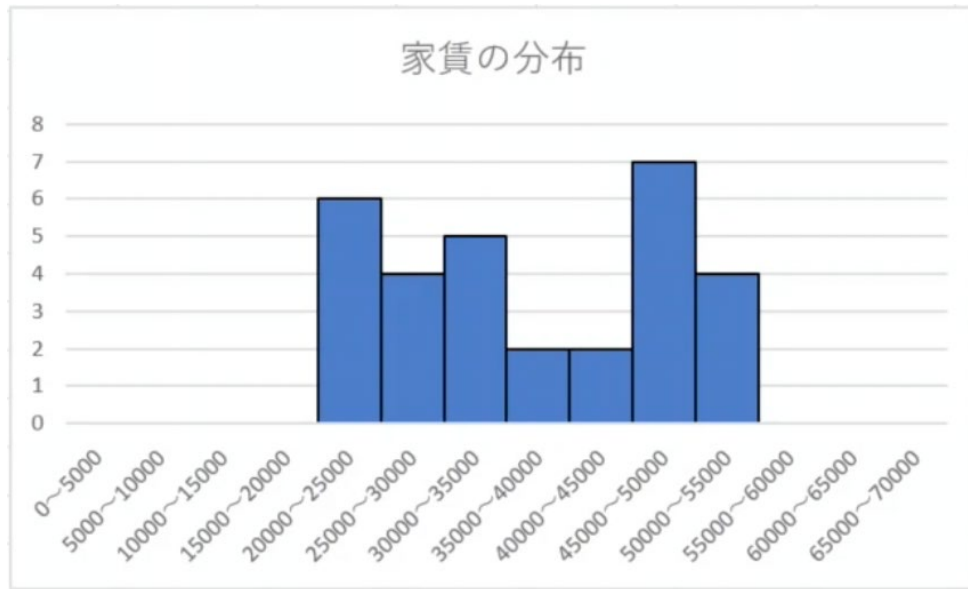


図 74 ヒストグラム

9.7.1 ヒストグラムとは

横軸に階級、縦軸に度数（該当する件数）を取って描いたグラフです。階級を規定するデータの所属する区間のことをビン(bin)と言います。ビンの幅が狭いと細かな表現が得られますが、各ビンに含まれるデータが少なくなり分布を掴むことは難しくなります。他方でビン幅が大きいと、分布の細かな状況が分からなくなります。

9.7.2 データの作成

家賃のデータを参照して、指定された範囲に入るデータの件数を数えます。

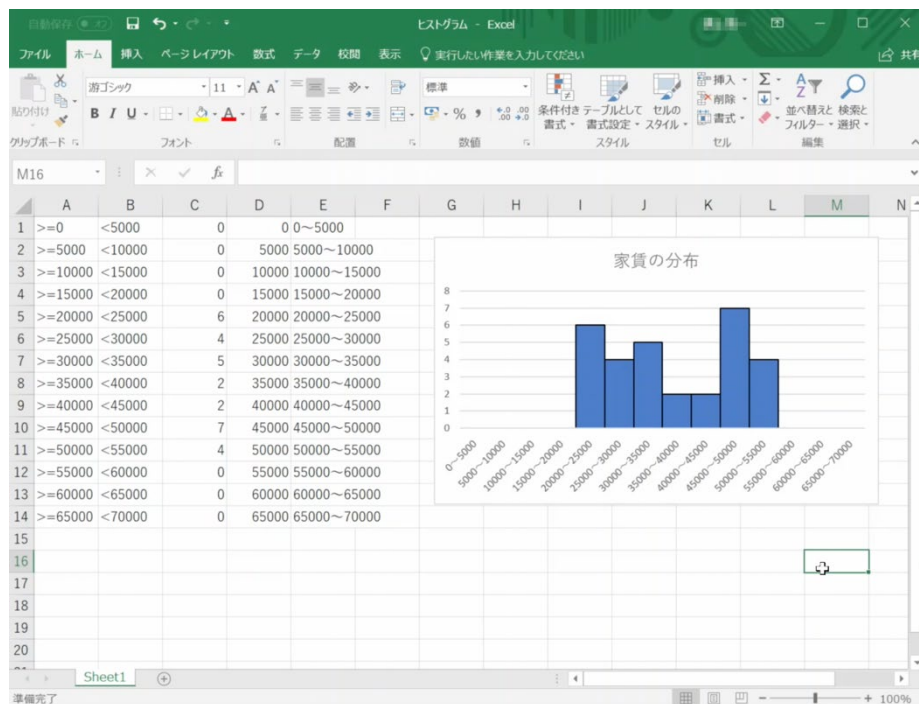


図 75 ヒストグラムのデータ作成

この例では新しいシート(Sheet2)をヒストグラムのデータ作成用に用いています。

- A列に範囲の下限を与える式（A1だと“>=0”）を、
- B列に範囲の上限を与える式(B1だと“<5000”）を書き、
- C列については例えばC1セルの内容を
`=COUNTIFS(データ!D2:D31,Sheet2!A1,データ!D2:D31,Sheet2!B1)`
 として、条件に合うデータの個数を数えています。

- D列には範囲の下限を数値だけ作成し、
- E列は軸ラベル用の文字列を
`=D1&"~"&(D1+5000)`
 （E1セルの内容の例）として作成しています。

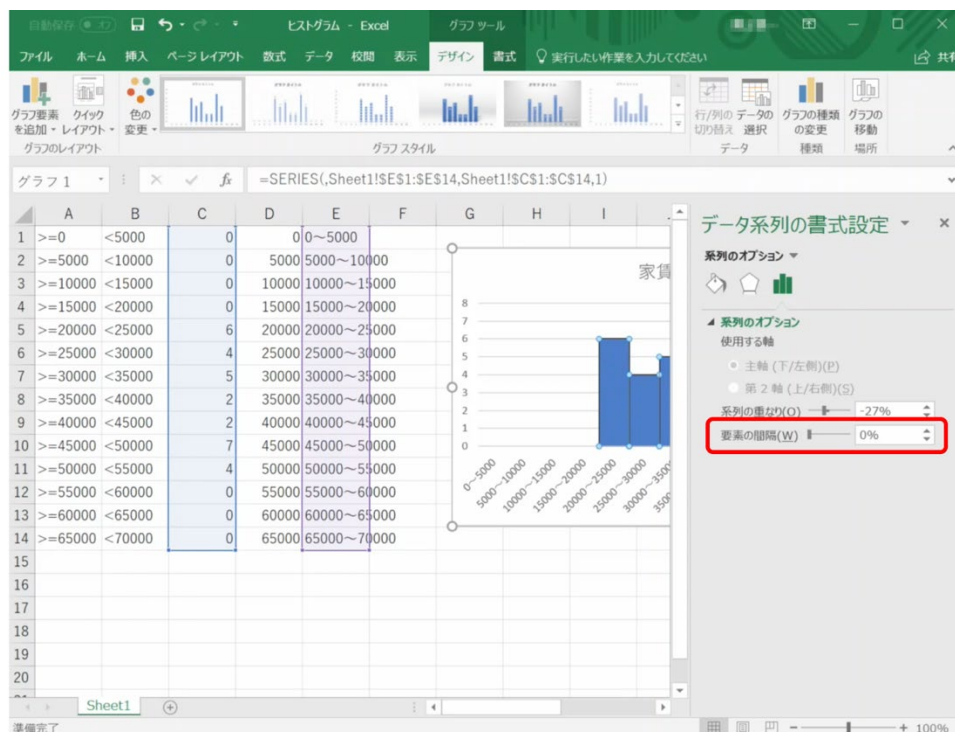
9.7.3 グラフのプロット

1. C列の計算結果を選んで縦棒グラフでプロットします。
2. データ系列を選んで右クリックし、「データ選択」を選び、横（項目）

軸ラベルの編集を選んで、軸ラベルとしてE列の値を指定します。



3. ヒストグラムは縦棒の間を空けません。データ系列を選んで右クリックし、「データ系列の書式設定」を選び、系列のオプションで「要素の間隔」を「なし」にします。また枠線の色を適宜設定します。



演習 32. 家賃の分布のヒストグラムを作成しなさい。

- データの縦軸のとり値の範囲は桁数は何桁もかわる広範なものであること

- 横軸の西暦年とともに、縦軸のデータの値の変化そのものが大きくなること

このため、線形目盛を用いた (b) のグラフでは左側のプロットがほぼ軸に張り付いてしまっていて変化の様子を読み取りにくくなっています。

これに対して、左側の図では

- 広範な範囲のデータを読み取りやすく表現できていること、
- データが直線的に並んでいること、

が読み取れます。人はデータがグラフ上で直線的に並んでいる場合は容易に直線的だと理解できます。このことから、グラフのプロットの仕方を工夫することで、データの特性が考察できるのです。

(a) のグラフのように、1 目盛が一定倍（ここでは 10 倍）になるようなグラフを対数グラフと呼びます。

10.2.2 線形目盛と対数目盛

通常、折れ線グラフや散布図ではプロットする値とプロットする位置は直線的な関係（値の一定の増加に対して、一定の量だけ位置だけ変化する）になっています。すなわち、データの値 x に対して、その位置 p を x の一次関数 $p = ax + b$ となる形で決めるのです。このような目盛の取り方を線形目盛(linear scale)と呼びます。

これに対して、値が一定倍になるごとに、一定の量だけ位置が変化する目盛を対数目盛 (logarithmic scale) と呼びます。これはプロットするデータの値の対数(logarithm)で位置を決めることによります。

対数とは、ある数 x に対して、定数 a を定め、

$$x = a^y \quad (7)$$

を満たすような数 y の意味し、

$$y = \log_a x \quad (8)$$

と書きます。定数 a は対数の「底(base)」と呼びます。

底（定数 a ）として 10 を用いたものを「常用対数」と呼びます。この場合、

1, 10, 100, 1000 の常用対数はそれぞれ以下の表のようにそれぞれ 0, 1, 2, 3 となります。

$1 = 10^0$	$0 = \log_{10} 1$
$10 = 10^1$	$1 = \log_{10} 10$
$100 = 10^2$	$2 = \log_{10} 100$
$1000 = 10^3$	$3 = \log_{10} 1000$

すなわち、対数と言うと難しく聞こえますが、これは「データの桁数 - 1」を表すものだと考えればよく、中間的な値に対してもうまく値が変化するように桁数の考え方を拡張したものと理解すればよいでしょう。

この対数を用いてデータの値 x に対して、その位置 p を対数 $\log_{10} x$ の一次関数 $p = a \log_{10} x + b$ となる形で決めるのです。

対数関数 $\log_{10} x$ の x の値の範囲（定義域）は正の数でなければならず、 x は 0 や負の数を取ることができません。このため、対数目盛には「0」となる位置はありません。

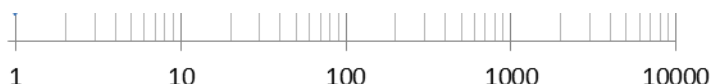


図 77 対数目盛 (0 は現れない)

10.2.3 片対数グラフ

先の図のようにグラフの片方の軸に対数目盛を用いるグラフを片対数グラフと呼びます。横軸とともに指数関数で変化するデータ

$$y = c a^x \quad (9)$$

については、両辺の対数¹³をとると

$$\log y = \log c + x \log a \quad (10)$$

となり、 $\log y$ が x の一次関数となることから、片対数グラフでプロットすることにより、直線となることが分かります。グラフの傾きは指数 a の対数となります。

10.2.4 両対数グラフ

横軸と縦軸の両方に対数目盛を用いるグラフは両対数グラフと呼ばれます。

¹³ 以下、底の値には依存しない議論ですので対数の表記で底を省略しています。

横軸のべき乗に比例して変化するデータ

$$y = c x^a \quad (11)$$

については、両辺の対数をとると

$$\log y = \log c + a \log x \quad (12)$$

となり、 $\log y$ が $\log x$ の一次関数となることから、両対数グラフでプロットすることにより、直線となることが分かります。グラフの傾きがべき乗の指数 a を表します。

10.3 Excel での対数グラフの作成

10.3.1 片対数グラフの作成

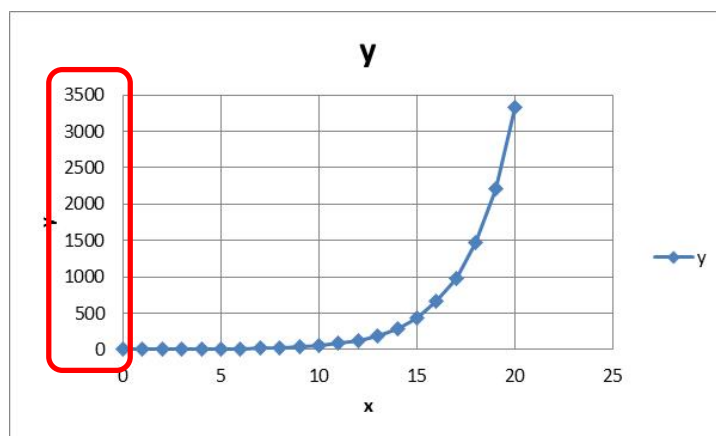
まず図 77 に示したようにデータを設定します。A 列は 0 から 20 までの数字が、B 列の 3 行目以降は一行上のセルの値の 1.5 倍となる式が設定されています。

	A	B	C
1	x	y	
2	0	1	
3	1	1.5	
4	2	2.25	
5	3	3.375	
6	4	5.0625	
7	5	7.59375	
8	6	11.39063	
9	7	17.08594	
10	8	25.62891	
11	9	38.44336	
12	10	57.66504	
13	11	86.49756	
14	12	129.7463	
15	13	194.6195	
16	14	291.9293	
17	15	437.8939	
18	16	656.8408	
19	17	985.2613	
20	18	1477.892	
21	19	2216.838	
22	20	3325.257	
23			

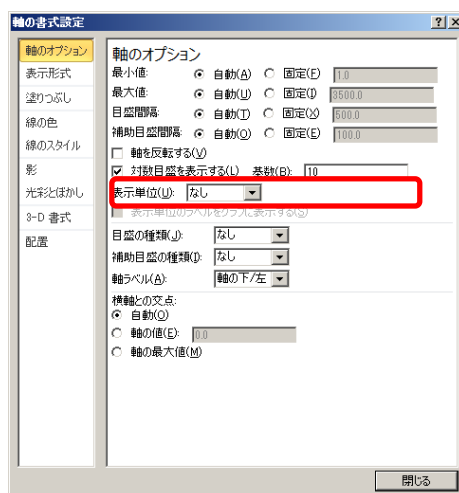
	A	B
1	x	y
2	0	1
3	1	=B2*1.5
4	2	=B3*1.5
5	3	=B4*1.5
6	4	=B5*1.5
7	5	=B6*1.5
8	6	=B7*1.5
9	7	=B8*1.5
10	8	=B9*1.5
11	9	=B10*1.5
12	10	=B11*1.5
13	11	=B12*1.5
14	12	=B13*1.5
15	13	=B14*1.5
16	14	=B15*1.5
17	15	=B16*1.5
18	16	=B17*1.5
19	17	=B18*1.5
20	18	=B19*1.5
21	19	=B20*1.5
22	20	=B21*1.5
23		

図 78 指数関数で増加するデータの設定
(左のシートは値、右のシートは数式で表示しています)

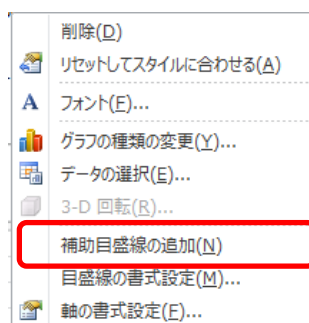
次にこのシートの A2:A21 を横軸に、B2:B21 を縦軸にマーカーを線で結ぶ散布図を描きます。系列名、軸ラベルなどは必要に応じて設定します。

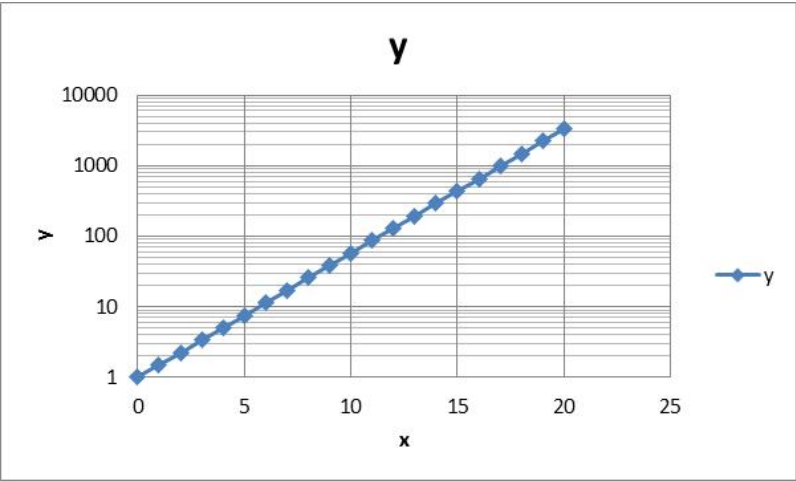


このグラフの縦軸を選択し、右クリックして、軸の書式設定を選び、「対数目盛を表示する」にマークします。



もう一度、縦軸を選択し、右クリックしてメニューから「補助目盛線の追加」を選びます。





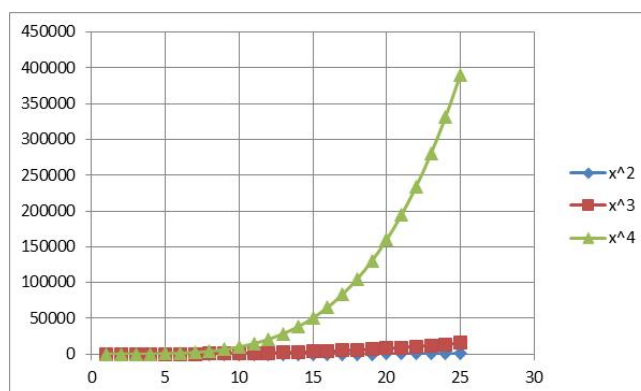
10.3.2 両対数グラフの作成

今度は両対数グラフを作成します。両対数グラフの効果を確認するために、横軸の変数 x の 2 乗、3 乗、4 乗を計算する表を作成します。

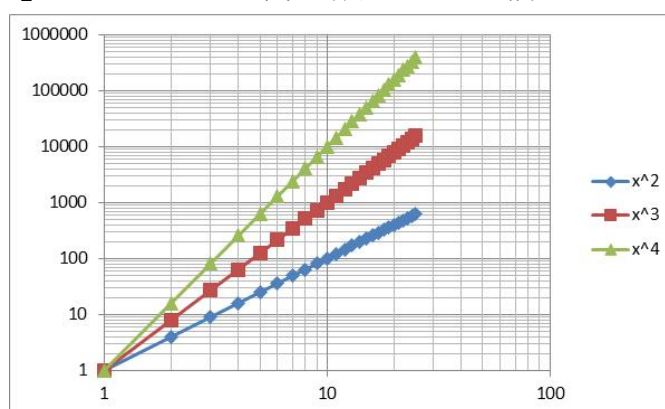
	A	B	C	D
1	x	x^2	x^3	x^4
2	1	1	1	1
3	2	4	8	16
4	3	9	27	81
5	4	16	64	256
6	5	25	125	625
7	6	36	216	1296
8	7	49	343	2401
9	8	64	512	4096
10	9	81	729	6561
11	10	100	1000	10000
12	11	121	1331	14641
13	12	144	1728	20736
14	13	169	2197	28561
15	14	196	2744	38416
16	15	225	3375	50625
17	16	256	4096	65536
18	17	289	4913	83521
19	18	324	5832	104976
20	19	361	6859	130321
21	20	400	8000	160000
22	21	441	9261	194481
23	22	484	10648	234256
24	23	529	12167	279841
25	24	576	13824	331776
26	25	625	15625	390625

図 79 べき乗で増加するデータの設定
(左のシートは値、右のシートは数式で表示しています)

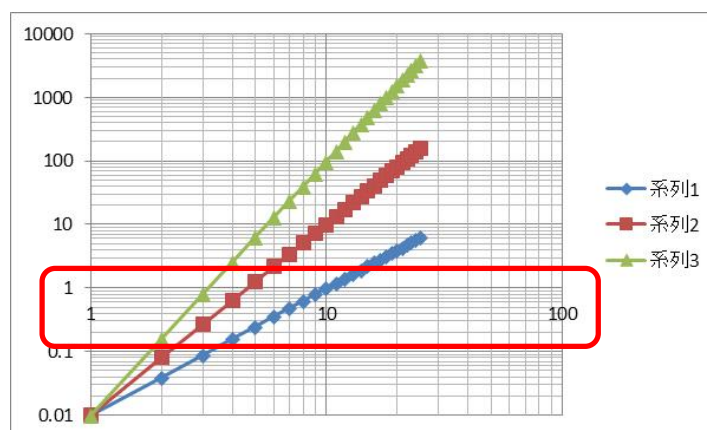
このデータに対して「複数系列」の散布図を以下のように作成します。



この図について、横軸、縦軸共に「対数目盛を表示する」を選び、「補助目盛線の追加」を選ぶことで両対数グラフが描けます。

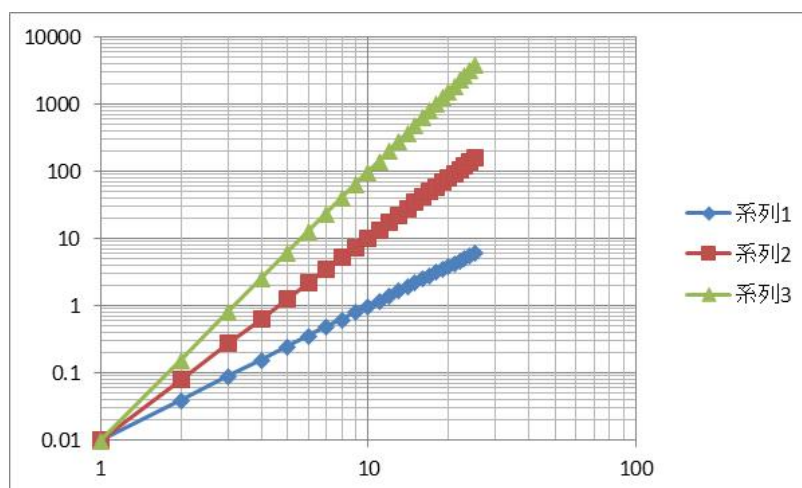
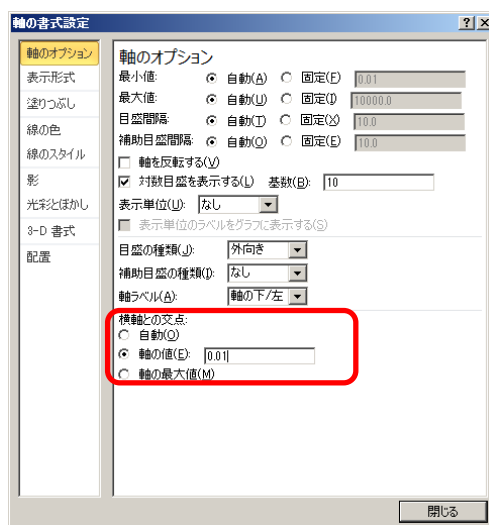


縦軸が1より小さい値を取る時には両対数グラフを描くと軸がグラフエリアの中に描かれてしまいます。



これをグラフエリアの下に移すには「縦軸」を選び、書式設定の中で横軸との交点を設定します¹⁴。

¹⁴ 横軸の位置を調整するために縦軸の書式を設定しなければならないので見つけないくいです。



演習 33. 皆さんの周りで対数グラフを用いて表示されている例を探してみてください。

演習 34. 指数関数的な変化をするデータやべき乗で変化するデータを探し、それぞれ適切な対数グラフでプロットしてみてください。

10.4 グラフにおける単位の表記

グラフの軸に単位などを表記することは学術的な文章では必須の事項ですが、自然科学の論文などでは軸ラベルとして

$$V/10^3 \text{mh}^{-1} \quad (13)$$

といった表現が用いられます(文献[1]の図 9.3 に示された例)。すなわち、

速度を表す変数 V をその単位 10^3mh^{-1} (キロメートル毎時, ローマン体で記述されていることに注意)で割算したものである, という表記です。これは速度 V に対して, 速度を表現する尺度である単位 10^3mh^{-1} で割ることにより, 次元のない数値が得られ, これをグラフの軸に用いている, ということを表しています。

すなわち

- 速度 V は「長さ L 」／「時間 T 」の次元を持つ物理量
- 単位 10^3mh^{-1} も「長さ L 」／「時間 T 」の次元を持つ物理量のものさし

から

$$\frac{V}{10^3\text{mh}^{-1}}$$

という「ものさしの何倍か」, という無次元の数値を得て, グラフにはこの値を示しているという考え方です。

参考文献

- [1] 木下是雄：理科系の作文技術, 中央公論社 (1981)

11. 構造化された文書の作成

11.1 学習の目的

- 論文やレポートの文章がどのように構造化されているかを学びます。
- Word でのパラグラフの扱いと章立ての方法を学びます。
- Word での参考文献の記載やその引用の方法を学びます。
- Word での文章の校閲の方法を学びます。

11.2 論文やレポートの構造

論文やレポートは論理的に物事を表すために以下のような構造を持たせて書かれています。

- **題目、著者、日付**
- **要約**
- **章立てと見出し**、章、節などの形で章立てが行われ、しばしば見出しに番号が振られています。番号は前後の章を他の箇所で参照する際に使います。著者が参照しなくても、その文章を引用したり、批評したりする他の著者によって使われることもあります。章や節の見出しのフォーマット（フォントや大きさ、行の揃え方）は文章内で一貫した表記法がとられます。本文以外に謝辞や付録が付けられることがあります。
- **パラグラフ**、学術的な文章ではまとまった内容の文章のかたまりを「**パラグラフ**、**paragraph**」として表記します。パラグラフの冒頭を下げたり（**字下げ**、**インデント**、**indent**）、パラグラフの間の行間を長めにとったりすることで、視覚的にパラグラフであることを分かりやすくします。

- **箇条書き**，並列する内容，順序のある内容（項目，アイテム，item）を分かりやすくするために箇条書きが用いられます。項目の順序を明示的に示したり，項目の参照を容易にしたりするために通し番号が付されることがあります。
- **注**，本文の内容を補足するものではあるが，本文の流れを妨げたくない場合に脚注や文末注が用いられます。
- **参考文献とその引用**，論文，レポートではその情報源を明示することで，議論している内容の論拠を示します。通常，文章末に参考文献のリストを付け，文章内でそれを引用する形で書きます。文献や引用は読者がその情報源に的確に辿りつけるだけの情報（書誌情報）が含まれることが求められ，標準的な記法が提唱されていたり，学術雑誌等で定められた書式に従う必要があったりします。
- **図や表の挿入とその参照**，本文中に図や表を挿入し，**図番号と説明（キャプション，caption）**，**表番号と標題（タイトル，title）**が付されます。図番号と説明は図の下に，表番号と標題は表の上に入れられることが多いです。本文中では図や表を図番号，表番号で参照します。
- **数式**，数式を本文中に挿入したり，独立した行として挿入したりします。

演習 35. 文献例の構造を探る

指示された文献についてどのような構造と表現上の書式が使われているかに留意して読んでください。

11.3 人がすること，コンピュータに任せること

メールなどで行事や会議の開催日を通知するときに日付に曜日をつけて書きますが，しばしば曜日が誤って付けられ，後で訂正されることがあります。なぜ，曜日を誤って書いてしまうのでしょうか，どうすれば誤りはなくせるのでしょうか？誤る理由は，日付と曜日のどちらもを人間が直接キーボードで入力しているからだと推察されます。他方で，日付（年月日）が決まれば曜日は自動的に決まるもののはずです。もしそうなら，人

は年月日のみを入力し、曜日はコンピュータに補ってもらう方がいいのではないのでしょうか。

レポートなどを書いていると、章や節に番号を振ったり、振られた番号を用いて文中で他の章や節を番号で指示したりします。番号を振ることは単純な作業ですが、文章の執筆中は章を追加したり、削除したりします。その度に人手で番号を振り直し、章番号を引用している箇所を修正したりしなければなりません。短い文章ではすべて手作業でやることも考えられますが、少し量が多くなればこの作業は手間がかかり、修正漏れも発生するでしょう。それならば、**章に番号を振ったり、引用している章番号を常に正しい値に維持したりすることはコンピュータに任せてしまう**、という考えに至ります。

またレポートでは見出しなどを付けますが、番号を付けるだけでなく、目立たせるために本文に比べ文字の書体を変えたり、大きめにしたりします。これについても、**1) 文章には「見出し」としての指定だけを行って、2) 「見出し」として指定された個所の表現方法は別途、設定します**。これにより、見出しの表現を統一的に調整することをコンピュータに任せます。

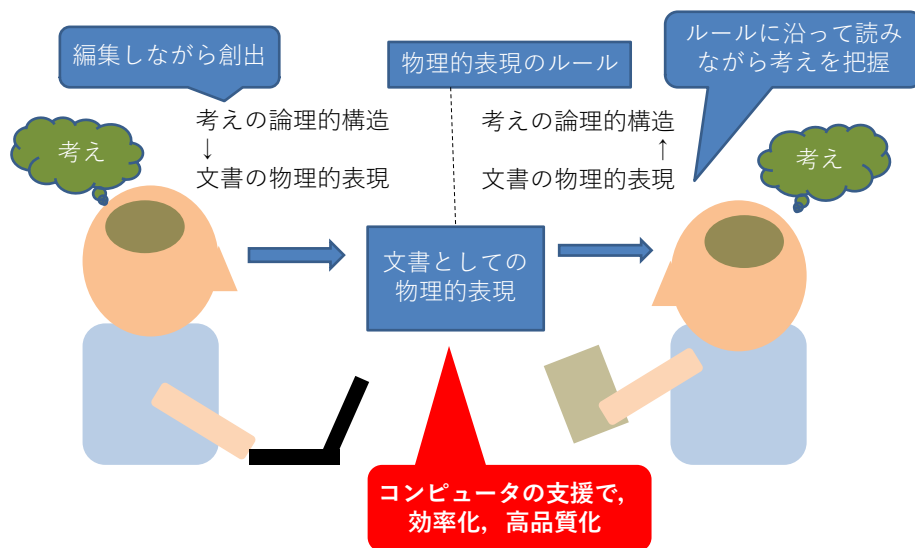


図 80 「考え」の文章化と文章からの「考え」の獲得

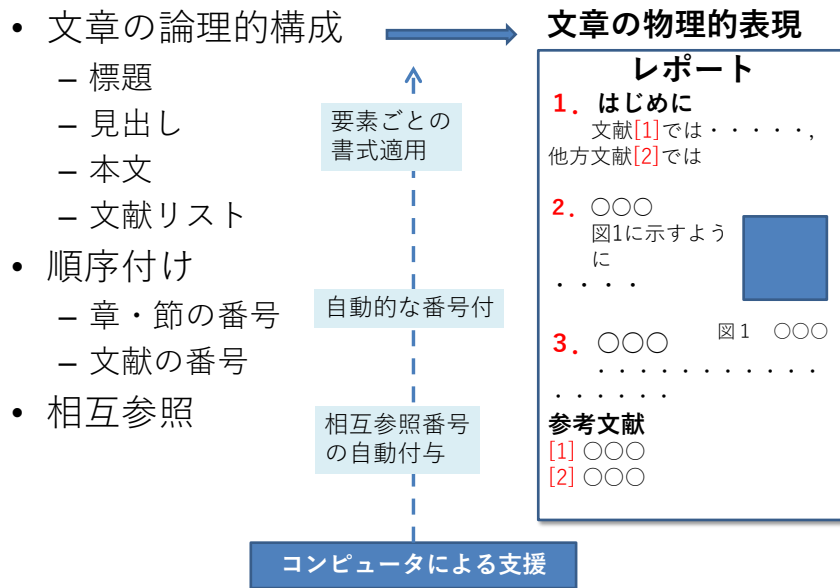


図 81 文章の論理的構成から物理的表現への変換

11.4 スタイルとテンプレート

Word には標題，見出し，本文，参考文献リストといった文章を構成する個々の要素に対して，効果的，効率的に書式を自動的に付与する仕掛けがあり「**スタイル**」と呼ばれます。そして，一連のスタイルを収めたファイルを「**テンプレート**」と呼びます。例えば新規文書を作成するときには「空白の文書」を選ぶことが多いと思いますが，これは「**空白の文書**」という**テンプレート**を適用して，文書ファイルを作成することを意味します。

11.4.1 読みやすさを重視したスタイルの必要性

学術や業務などの目的で書かれるレポートなどの文章は**読みやすさに配慮することが重要です**。Word では書体（フォント）をさまざまに選択することが可能ですが，

- 通常の文章は明朝体で，
- 見出しなど強調する箇所はゴシック体で書くことをおすすめします。

英文については明朝体やゴシック体をそのまま使うのではなく英文用

のフォントを使うと読みやすくなります。和文と英文が混在する場合は似た書体を組み合わせて使います。例えば、以下のような組み合わせになります（和文フォントのうち、游明朝、游ゴシックは Windows7 では標準では導入されていません）。英文フォントについては文献[9]で推奨されているものを挙げました。

表 10 レポートで使うフォント

和文	和文フォント	和文フォント	英文フォント
明朝体	游明朝	MS 明朝	Times New Roman
ゴシック体	游ゴシック	MS ゴシック	Segoe UI

文字の大きさはレポートでは 10.5 ～ 12 ポイント程度で選びます。行間は文字の大きさの 0.5 ～ 1 倍に選ぶようにします[9]。

文章の特定箇所を強調したり引用箇所を他と区別するなどの目的で、テキストを太字や斜体（イタリック体）にしたり下線を引くことがあります。変更したい箇所を選択し、「ホーム」リボン上の「太字」「斜体」「下線」の各ボタンを選択します。それぞれ、「Ctrl+B」「Ctrl+I」「Ctrl+U」のショートカットが利用できます。

表 11 代表的なショートカット

ショートカット	
太字	Ctrl + B
斜体	Ctrl + I
<u>下線</u>	Ctrl + U

11.4.2「空白の文書」のスタイルの指定

「空白の文書」をテンプレートに新しい文書を作成した場合、「空白の文書」で定義されているスタイルが利用できます。よく使うスタイルはホームで表示されるリボンの「スタイル」のなかのクイックスタイルギャラリーに表示されていますが、利用できるスタイルの一覧を表示するには、リボンのスタイルの右下の矢印をクリックしてください。

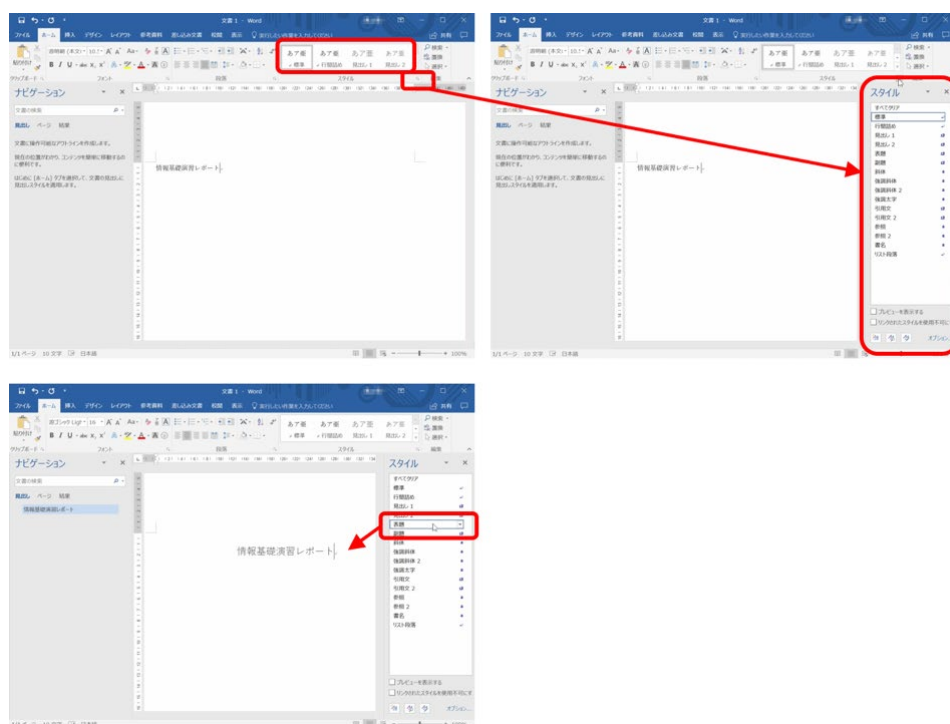


図 82 スタイルの適用

11.4.3 レポート用テンプレート

「空白の文書」で設定されているスタイルは章見出しの番号付けなどは必ずしも大学でのレポートに適したものとはなっていません。レポートの作成を支援するために

レポート論文用表紙なし 20160402.dotx

というテンプレートを用意しました。このテンプレートの使い方については、資料「レポート論文用表紙なしテンプレートの使い方」を参照してください。

11.5 パラグラフの書き方

Word では 1 つのパラグラフの中では改行せずに文をつづけます。スタイルを「標準」あるいは「本文」とし、字下げは文書の上にあるルーラーで調整します。字下げの整形のために文頭に空白を挿入することは望ましくありません。なぜなら、先頭に文を追加するなどして、パラグラフの構成を変更したりする場合、空白は段落中に残ってしまい、これを消すことを見落とす可能性があるからです。

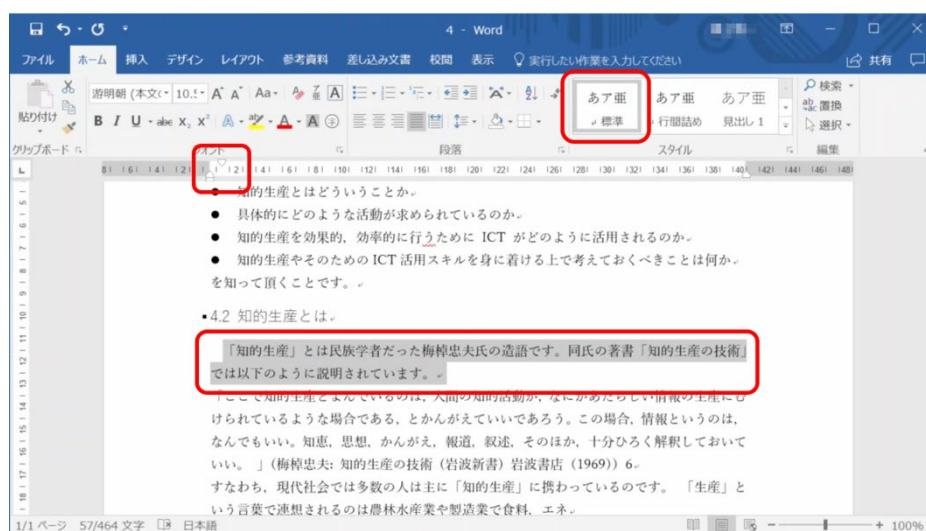


図 83 パラグラフ先頭の字下げ

- ルーラーが表示されていない場合、表示リボンにある「ルーラー」にチェックを入れます。ルーラーの左右にある三角・四角のマーカーを使ってインデント（字下げ）を調整します。
- インデントを適用するには、該当のパラグラフにカーソルを置きます。複数のパラグラフを選択することもできます。
- 左側のマーカーのうち、四角（□）のマーカーはパラグラフ全体の左側のインデント、下向きの三角（▽）はパラグラフの 1 行目のインデントを調整します。
- インデントを微調整したい場合は、alt キーを押しながらマーカーを移動させます。



図 84 ルーラーを使ったインデントの調整

11.6 章立ての見出しの書き方

章立ての見出しは以下の手順で書きます。

1. 見出しを書きます。
2. 見出しのスタイルをそのレベルに合わせて「見出し 1」、「見出し 2」など選びます。
3. 段落リボンからアウトラインボタンを選び、見出し番号の形式を選びます。**アウトラインボタンのうち「見出し」という言葉が含まれるものを選びます。**そうでないものは箇条書き用のものです。

なお、Word の「空白の文書」として設定されている見出しの番号形式にはレポートでしばしば用いられる 1. や 2.3 といったものが設定されていません。これについては、新しいアウトラインボタンを押して現れるメニューから「新しいアウトラインの定義」を用いて調整します。

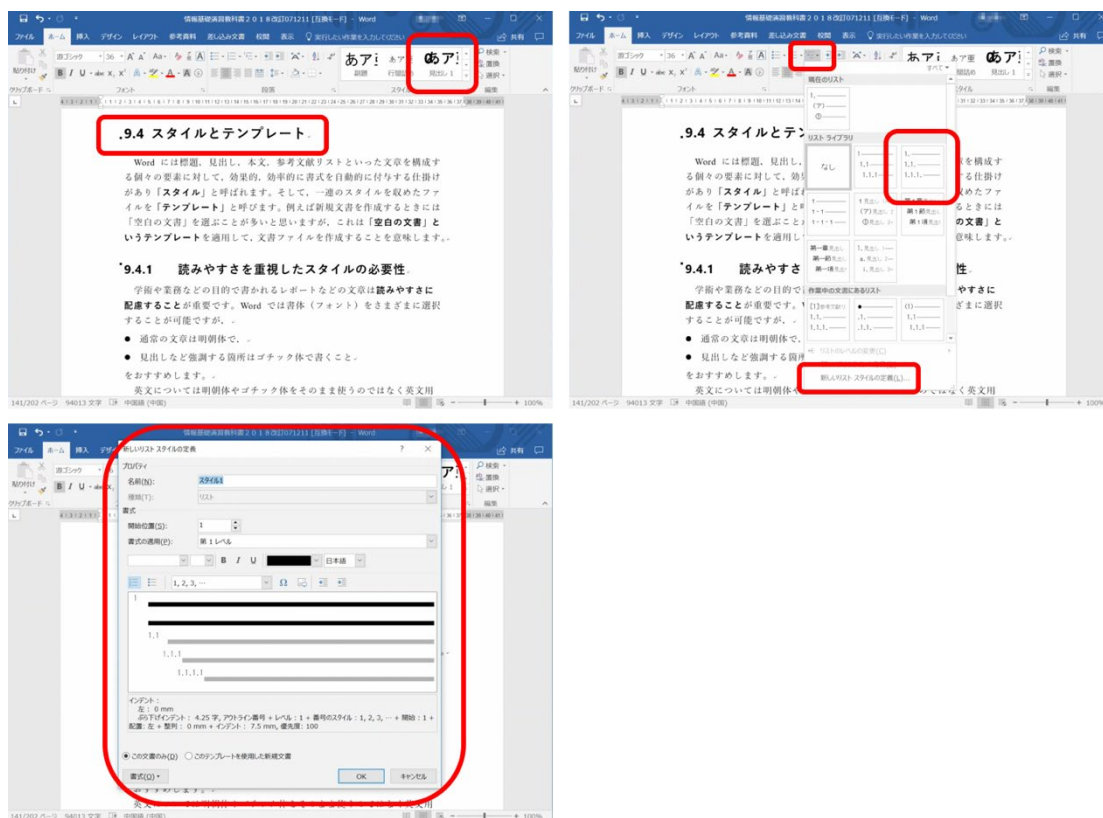


図 85 見出し用スタイルの調整

また、「参考文献」は見出しに番号をつけないことが多いですが、この場合は、参考文献という見出しに対して、アウトラインボタンで「なし」を選んでください。

演習 36. パラグラフと見出しの設定

演習課題用文章に対して、指示に従ってパラグラフと見出しをつけなさい。

11.7 箇条書き・脚注・ページ設定

11.7.1 箇条書きの活用

文章中に箇条書きを用いるには、箇条書きにしたい範囲を選択し、「ホーム」リボンにある「箇条書き」ボタンをクリックします。番号付きの箇

条書きにする場合は、その隣の「段落番号」を使います。

箇条書きの各項目のレベル（階層）を変更するには、変更する項目を選択し「インデントを増やす」（または「インデントを減らす」）をクリックします。

番号付きの箇条書きで開始番号を 1 以外に変更したい場合、「段落番号」の▼記号から「番号の設定」を選択し、表示された画面で開始番号を入力します。

箇条書きの各レベルで用いる数字や記号の書式を変更する場合、「アウトライン」に表示される候補から選択するか、「アウトライン」から「新しいアウトラインの定義」をクリックし、各レベルの番号書式を選択します。

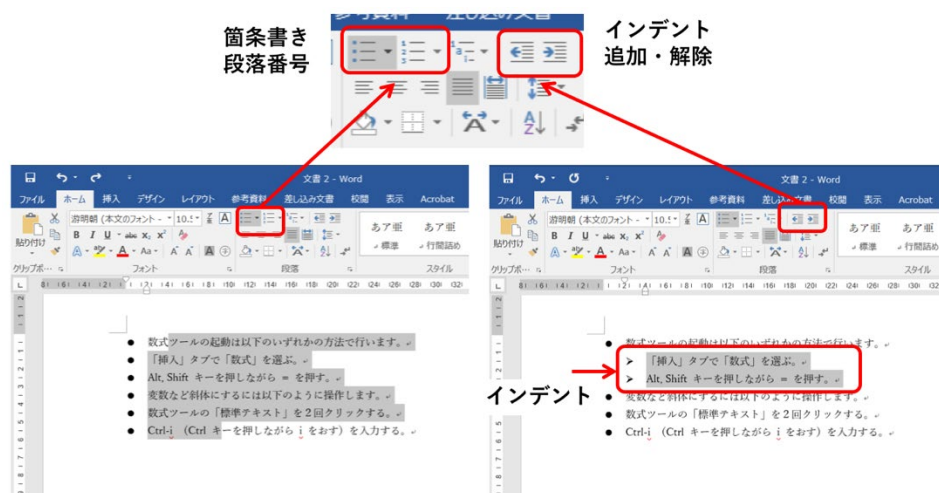


図 86 箇条書きとインデントの変更

11.7.2 脚注の作成

脚注を用いるには、本文中で脚注を付けたい箇所にカーソルを移動し、「参考資料」リボンの「脚注の挿入」をクリックします。本文中には脚注番号が自動的に上付き文字で表示されます。

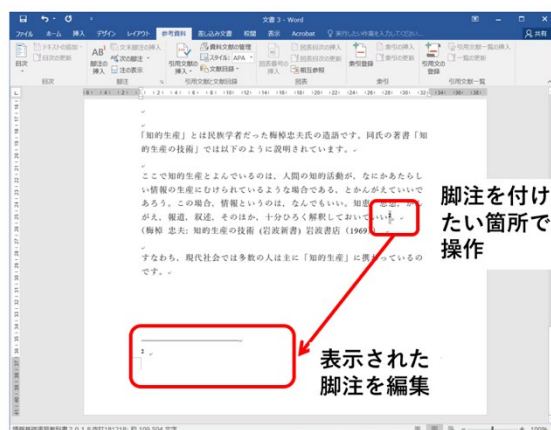


図 87 脚注の作成

11.7.3 余白の変更

- ページの上下左右にある余白の幅を変更するには、「レイアウト」リボンで「余白」を選択し、表示されるレイアウトの中から選ぶか、「ユーザー設定の余白」を選択して余白の幅を指定します。
- 余白の変更はルーラーを使っても可能です。変更するにはルーラー上の白と灰色の境目をドラッグします。

11.7.4 ページ番号の挿入

文書にページ番号を挿入するには、「挿入」リボンで「ページ番号」をクリックするとページ番号を挿入する位置が選択できます。開始番号を1以外に変更したい場合は、「ページ番号」から「ページ番号の書式設定」を選択し、番号の書式や開始番号を設定します。

11.7.5 改ページ

ページの途中で次のページへ移りたい場合、「挿入」リボンから「ページ区切り」を選択します。

11.8 アウトライン表示の活用

11.8.1 アウトライニング

学術的な文章のように、多数の資料などを参照しながら論理立った文章

を書くことは決して容易ではありません。文章を執筆するための素材から、文章の構成の骨格（アウトライン）を考え、これをパラグラフとしてまとめていく形で文章を作成することは、文章作成のためのよい方法です。

11.8.2 アウトライン表示の利用

Word ではアウトライン表示を使うことでこの作業の効率を高めることができます。「表示」メニューで現れるリボンの「文書の表示」から、「アウトライン」を選びます。

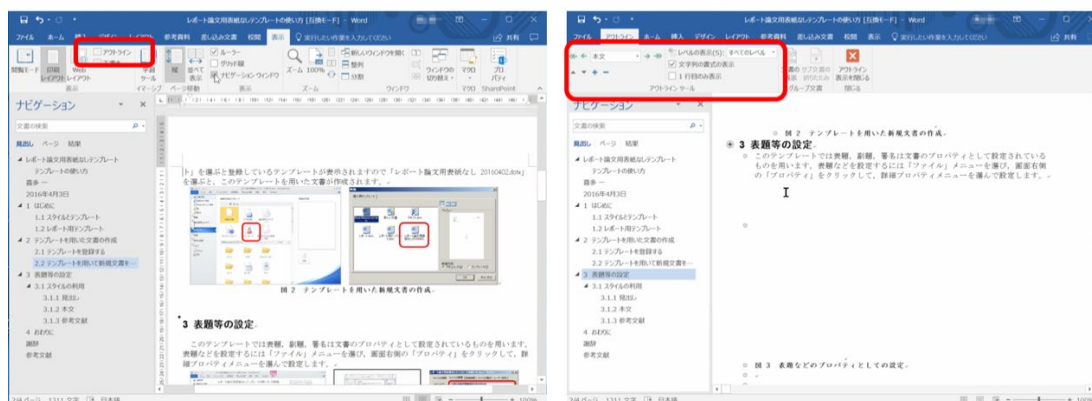


図 88 アウトライン表示

- アウトラインツールの「+」「-」で章や節を展開したり折りたたんだりできます。
- ▽や△で選択した章や節全体を上下に移動することができます。

11.9 参考文献の記載方法とその引用

11.9.1 2 種類の引用

学術的な文章の作成では多くの資料などを利用します。このため使用した資料については文章の中でそれらを引用し、出典を明記するなど適正な取り扱いをしないとはいけません。

引用には 2 種類の方法があります。

1. 参照した文献の書誌情報を示す形での引用。Citation と呼ばれます。
2. もとの文献から必要な個所を原文のまま文章内に引用する方法。

Quotation と呼ばれます。

両者で共通することは出典を明記することです。以下の参考文献の記載方法に従ってください。

これに加えて、必要な個所を原文のまま文章内に引用することについては著作者の許諾を得ることなく引用ができますが、以下の要件を満たすことが求められます。(文献[13]より引用)

1. すでに公表されている著作物であること
2. 他人の著作物を引用する必然性があること
3. 引用部分が明瞭に区別されていること（引用部分に「 」をつけるなど）
4. 引用の範囲が必要最小限であること
5. 自分の著作物を主、引用する著作物を従としての主従関係があること
6. 原則として原形を保持して掲載すること
7. 原著者の名誉を侵害したり、原著者の意図に反した使用をしたりしないこと
8. 出所（出典）を明示すること

このうち、3. については以下のような表記が多く用いられます。

- 引用箇所の書体を変える。
- 引用箇所を括弧で囲む。
- 引用箇所を独立した段落で構成し、左右のマージンを本文より内側にする。

4.2 節では引用箇所は左右のマージンを 3 文字分、通常の記事より内側

に設定しています。以下に再掲します。

ここで知的生産とよんでいるのは、人間の知的活動が、なにかあたらしい情報の生産にむけられているような場合である、とかんがえていいであろう。この場合、情報というのは、なんでもいい。知恵、思想、かんがえ、報道、叙述、そのほか、十分ひろく解釈しておいていい。

(梅棹 忠夫: 知的生産の技術 (岩波新書) 岩波書店 (1969))

11.9.2 参考文献リストの書き方

ここでは参考文献は単純に番号付けされた箇条書きで実現しています。また「参考文献」という見出しを「見出し1」のスタイルで設け、特に番号は付さない形にしています。

参考にする資料には書籍、論文、Web サイトなど多様なものがあります。参考にしたものは必ず参考文献として掲載することと、その記載（書誌情報）については読者が適切にその内容を確認できるための情報が得られることが基本です。参考文献の記載方法については「科学技術情報流通技術基準」[12]を参照するとよいでしょう。

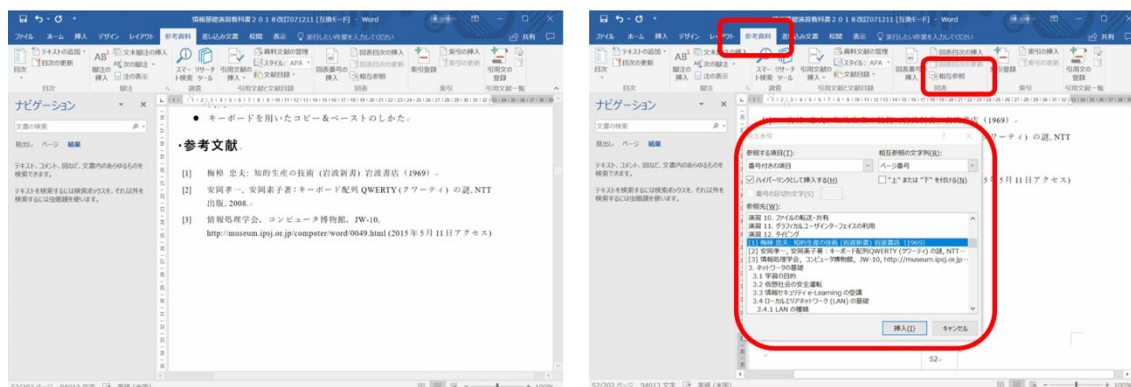


図 89 参考文献リストの記載と参照

11.9.3 引用の書き方

本文中などで文献を引用する場合、文献リストの番号を挿入する必要があります。先のように番号付の箇条書きにしてあれば、「参考資料」というメニューのリボンにある「相互参照」で文献の番号を挿入できます。

11.9.4 引用などの更新

参考文献のリストに追加や削除があった時は該当箇所（文章全体なら「すべてを選択」）を選んで右ボタンをクリックし、「フィールドの更新」を選べば文献番号が付け直されます。

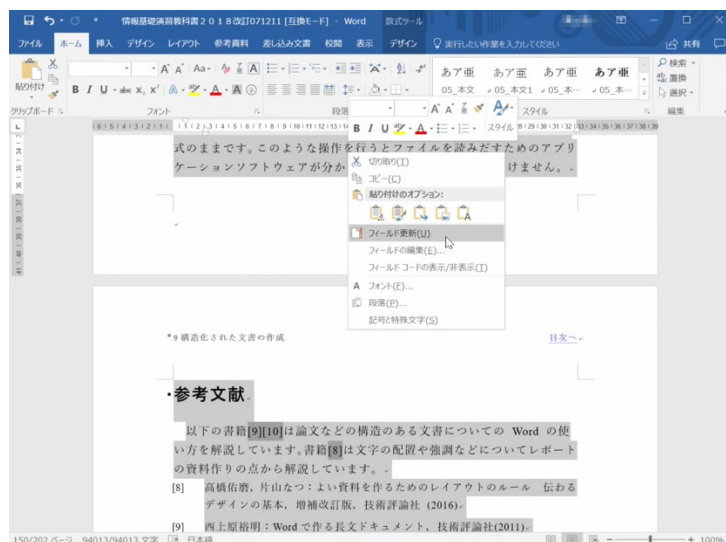


図 90 フィールドの更新による相互参照の更新

演習 37. 参考文献リストの作成と相互参照

演習課題用文章について、参考文献を箇条書き形式で作成し、引用箇所について相互参照で書き直しなさい。またフィールドの更新について操作を確認しなさい。

11.10 文章の校閲方法

11.10.1 文章の校閲とその他の Word の機能

文章は何度か書き直して仕上げて行きますが、その際のどこを修正したのかが分かることが望まれます。とりわけ、グループで文章を作成している場合や指導教員から指導を受ける場合など修正箇所を分かりやすくすることはコミュニケーションの質を高める上でも重要です。

Word では「校閲」というメニューに変更箇所の記録を取れる「変更履歴の記録」という機能があります。また、文章の特定の箇所に対して注釈

をつける「コメントの挿入」という機能があります。

- 変更履歴を残したい場合、「校閲」リボンで「変更履歴の記録」をクリックします。追加した箇所は下線、削除した箇所は取り消し線が色付きで表示されます。
- 変更箇所を本文に反映させるには、下線または取り消し線がある箇所です右クリックし「挿入を反映」を選択します。または、「校閲」リボンの「承諾」ボタンでも変更箇所を反映させることができます。
- 変更履歴の記録を終了するには、再度「変更履歴の記録」をクリックします。
- コメントを付けたい範囲を選択し、「校閲」リボンで「新しいコメント」をクリックします。
- 画面の右側にコメントの入力欄が表示されます。
- コメントを削除するには「校閲」リボンで「削除」を選択するかその▼を選択してコメントの削除範囲を指定します。

演習 38. 変更履歴の利用とコメントの付与

演習課題用文章について変更履歴を記録した上で、同文中で梅棹忠夫氏の文章をそのまま引用している箇所について、段落の左右のマージンを漢字2文字分程度下げることで引用を分かりやすくしてください。また、このようにしたことをコメントとして付けてください。

11.11 PDF 形式の文書の作成

Word で作成した文書はそのままでは Word のファイル形式に対応したソフトでなければ閲覧できません。Web など公開されている文書には PDF と呼ばれる形式のファイルが用いられ

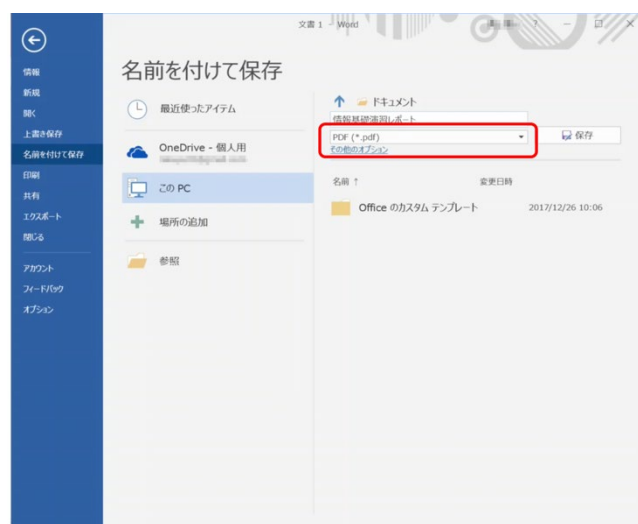


図 91 PDF 形式での保存

ます。レポートを提出する際にも PDF 形式での提出を求められることが少なくありません。Word で作成した文書を PDF 形式にするには以下の手順で行います。

- [ファイル]メニューを選び、[名前をつけて保存]を選択します。そしてファイルの種類として PDF を指定して保存してください。

注意 Windows ではファイルの形式に応じた拡張子が用いられます。Word のファイル(doc や docx などの拡張子がつけられています) の拡張子をエクスプローラーで pdf に変えるとアイコンが PDF のリーダーなどに変わりますが、ファイルの中身は変わらず、中身は Word 用のファイル形式のままです。このような操作を行うとファイルを読みだすためのアプリケーションソフトウェアが分からなくなるので行ってはいけません。

参考文献

以下の書籍[10][11]は論文などの構造のある文書についての Word の使い方を解説しています。書籍[9]は文字の配置や強調などについてレポートの資料作りの点から解説しています。

- [9] 高橋佑磨, 片山なつ: よい資料を作るためのレイアウトのルール 伝わるデザインの基本, 増補改訂版, 技術評論社 (2016)
- [10] 西上原裕明: Word で作る長文ドキュメント, 技術評論社(2011)
- [11] 相澤裕介: 論文・レポート作成に使う Word 2016 活用法, カットシステム (2016)
- [12] 科学技術情報流通技術基準 http://sti.jst.go.jp/sist/menu_purpose/index.html (2016 年 10 月 26 日アクセス)
- [13] 大学学習資源コンソーシアム (CLR): 大学学習資源における著作物の活用と著作権 (2017), http://clr.jp/servicemenu/guideline_201704.pdf (2017 年 12 月 19 日アクセス)

12. 文章への図表の挿入

12.1 学習の目的

Word を用いて学術的な文章に図表を挿入する際の基本として以下のことを学びます。

- Word での図表そのものの挿入の方法を知ります。
- 図や表を挿入する際の番号付けや参照のルールを知ります。
- Word での番号付きの図表の説明の挿入と番号の本文での引用の方法を知ります。
- Excel で作成した表やグラフ、PowerPoint で作成した図などを Word の文章に挿入する際に複数の方法があることを知ります。

12.2 Word の文章への図表の挿入

12.2.1 図の挿入について

図表を挿入するには「挿入」メニューで現れるリボンから選びます。

ブロック図などは「挿入」メニューのリボンから「図形」を選んで作成します。「画像」は jpeg 形式や png 形式などの画像ファイルの挿入に使用します。PowerPoint で作成した図や Excel で作成した表、グラフの挿入については後述します。

図表の周りの文章の回り込みについては、図や表を選んで右クリックして現れるメニューで調整します。



図 92 図の挿入



図 93 文章の回り込みの設定

12.2.2表の挿入について

表を新たに作成するには、「挿入」リボンで「表」を選択します。選択すると挿入する表の行と列の数が指定できます。行や列は表の作成後でも追加、削除ができます。

表のデザインは、「表ツール」リボン内の「デザイン」「表のスタイル」から選択できます。

表の罫線の書式を変更したい場合、表の任意の場所にカーソルを置いた状態で、「表ツール」リボン内の「デザイン」で「罫線」を選択します。表中で選択した範囲内の罫線の付け方を選択します。

「表ツール」リボン内の「レイアウト」にある「行と列」グループから挿入や削除などの該当の操作を選択します。

表の各セル内のテキストの配置や、複数セルの結合や分割なども「レイアウト」内でおこなうことができます。

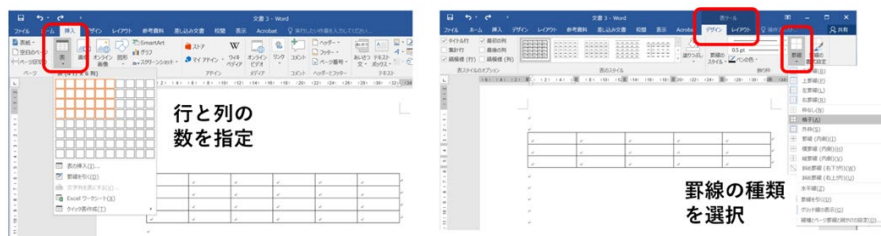


図 94 表の挿入

12.3 学術的文章での図表番号、タイトル、説明の付け方

12.3.1 番号やタイトル、説明の位置

学術的な文章での図や表は通常、以下のように扱います。¹⁵

- 図表には番号を振り、タイトルや説明を付けます。
 - 図の番号などは図の下側に付けることが通例です。
 - 表の番号などは表の上側に付けることが通例です。

12.3.2 本文での参照

文章に挿入した図や表は必ず本文で参照します。参照は「図 1」、「表 2」など番号で引用します。

12.3.3 出典の明記

図や表はそれ自体を他の文献から引用したり、あるいは公表されているデータをグラフにプロットしたりして作成することが多くなります。引用した図、表については出典を、データについては出典と、さらにプロットは著者自身が行ったことなどを明記します。

12.4 Word での図表番号の付与と相互参照

12.4.1 Word での図表番号の挿入

図表番号を挿入するには「参考資料」のメニューを選び、リボン「図表」から「図表番号の挿入」を選びます。表示されたウインドウで「ラベルを図表番号から除外する」にチェックを入れると、該当する図表番号の数字のみが挿入されます。

¹⁵ 理工系の領域で一般に行われる方法ですが、学問領域での慣習や論文誌などの発表対象での要項に沿う必要があります。

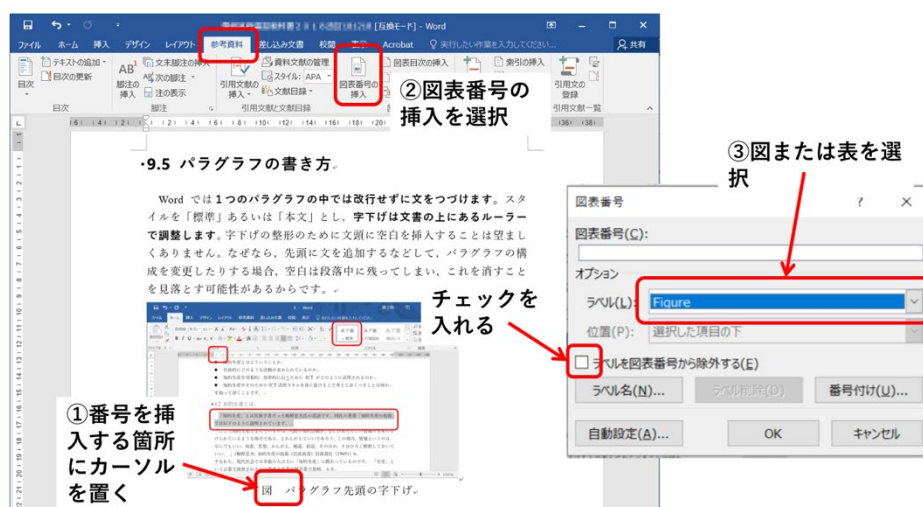


図 95 図表番号の挿入

12.4.2 Word での図表番号の参照

本文中で図表番号を参照するには「参考資料」のメニューを選びリボン「図表」から「相互参照」を選びます。「参照する項目」として「図」や「表」を選び、「相互参照の文字列」については「番号とラベルのみ」とすればよいでしょう。あとは参照先から挿入したいものを選びます。

表示されたウインドウ右上の「相互参照の文字列」の選択肢から「番号とラベルのみ」を選択すると、「図 1」「表 1」のような形式で図表番号が挿入されます。

本文の図表番号を Ctrl キーを押しながらクリックすると、該当する図表に移動します。

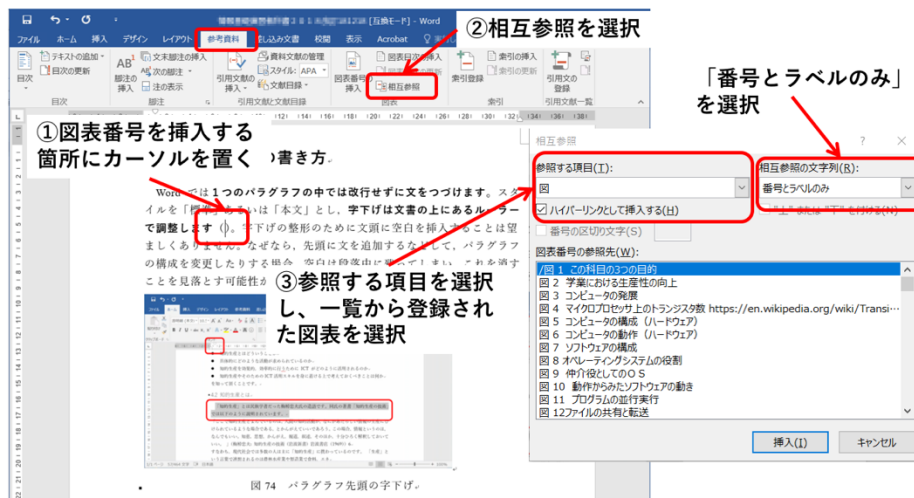


図 74 パラグラフ先頭の字下げ。

Word では1つのパラグラフの中では改行せずに文をつづけます。スタイルを「標準」あるいは「本文」とし、字下げは文書の上にあるルーラーで調整します(図74)。字下げの整形のために文頭に空白を追加することは望ましくありません。なぜなら、先頭に文を追加するラフの構成を変更したりする場合、空白は段落中に残った消すことを見落とす可能性があるからです。

④図表番号が挿入される
左の例では「図74」

図 96 図表番号の参照

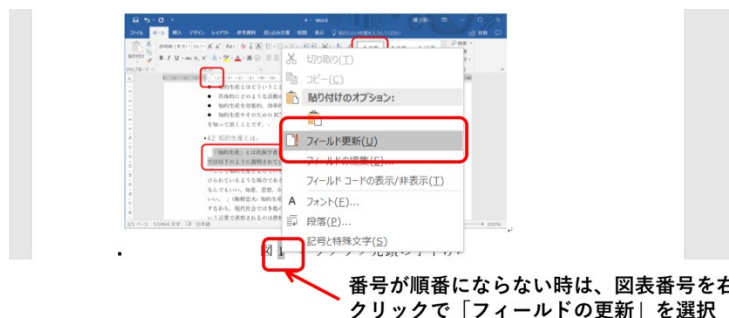


図 97 フィールドの更新

12.5 Word と Excel, PowerPoint の連携

12.5.1 データの形式と変換, コピーとリンク

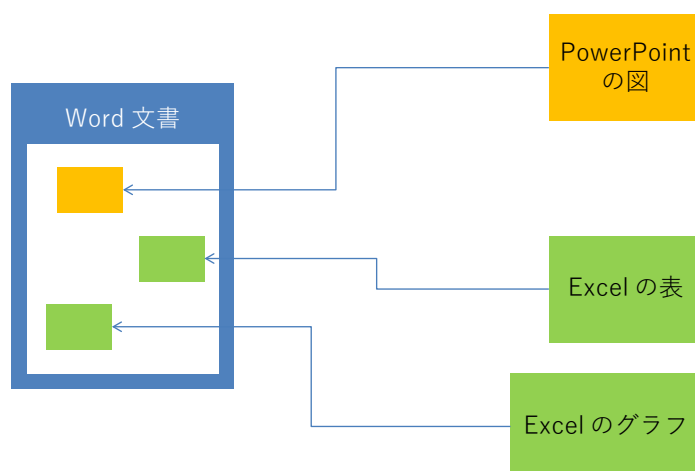


図 98 PowerPoint や Excel の情報の利用

Windows ではコピーしたり貼り付けたりするデータのことを「オブジェクト」と言います。オブジェクトとしてさまざまなデータの形式があります。Excel や PowerPoint で編集できるオブジェクトのほか、Word 内で扱えるオブジェクト、写真と同様の画像として扱うオブジェクトなどがあります。

オブジェクトをコピーして貼り付ける場合、

- もとのオブジェクトのまま扱うことや、
- 画像などの他の形式に変換して扱うこと

があります。

また、

- オブジェクトをコピーした上で貼り付ける場合と、
- もとのデータへのリンクとして扱う場合

があります。前者はコピーしてしまっているので、貼り付けたオブジェクトの編集はもとのオブジェクトと独立に行えます。後者はもとのオブジェクトにリンクしているので、もとのオブジェクトの変更が反映されます。

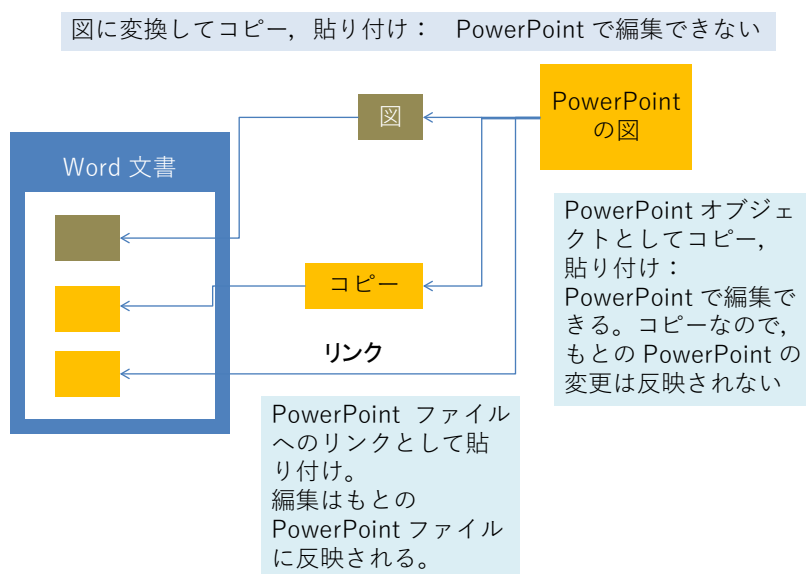


図 99 貼り付けるオブジェクトの形式と得失

12.6 PowerPoint の図の挿入

PowerPoint で作った図を挿入するには，PowerPoint の標準表示で左側に現れるスライドのウィンドウから，挿入したいスライドを選んでコピーします。次に挿入先の文書を編集している Word で

- スライドを図として挿入したい場合は，そのまま Word のカーソルがある位置で「貼り付け」をします。
- PowerPoint で編集できるオブジェクトとして貼り付ける場合は，ホームメニューのリボンの「クリップボード」から「貼り付け」の下の方の▼を展開し，「形式を選択して貼り付け」を選んで「Microsoft PowerPoint Slide オブジェクト」を選びます。
- このダイアログで「リンク貼り付け」を選ぶともとの PowerPoint ファイルへのリンクとして貼り付けます。

余白の調整は貼り付けたオブジェクトを選択し，右ボタンで現れるダイアログから「トリミング」の記号をクリックすることで行います。

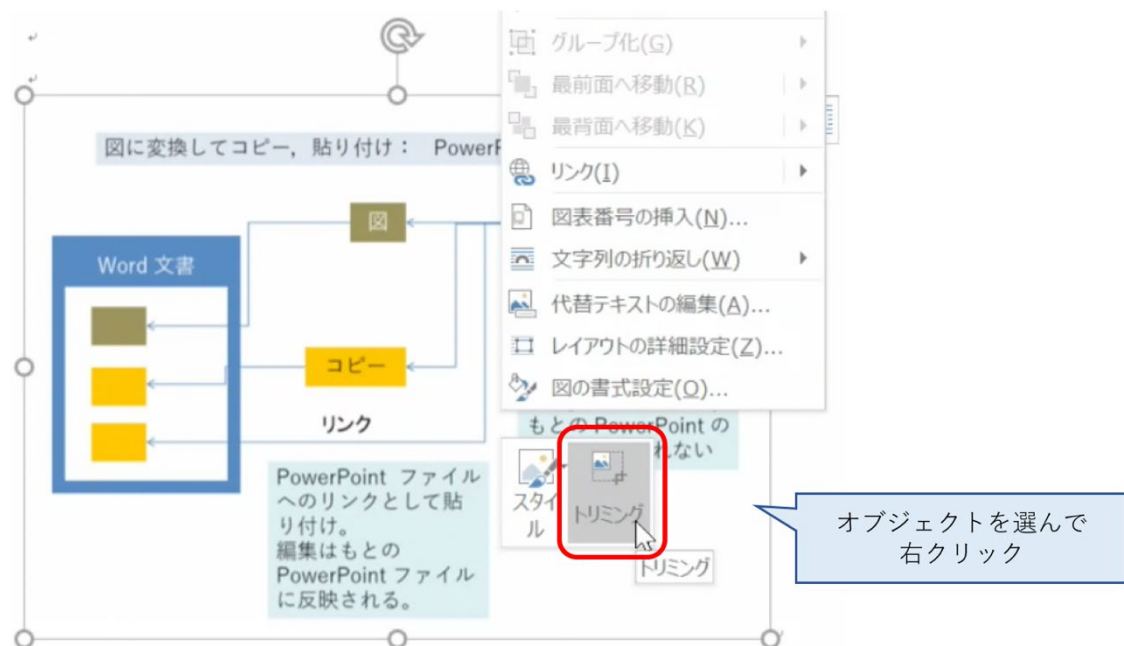


図 100 トリミングによる余白の調整

12.7 Excel の表やグラフの挿入

Excel の表やグラフも PowerPoint 同様、

- Excel で貼り付けたい部分をコピーし、
- Word で貼り付けます。
- 形式を選択して貼り付けることで Excel で編集可能なオブジェクトやもとのファイルへのリンクとして貼り付けることが可能になります。形式を選択して貼り付けるには、Word 上の該当箇所ですクリックし「貼り付けのオプション」から選択します。

13. 文書での数式の扱い

13.1 学習の目的

- 学術的な文章での数式の扱いについて学びます。
- 数式特有の変数などでの字体の選択について学びます。
- Word での数式の挿入方法について学びます。

13.2 表現の手段としての数式

高等学校でも数学や理科で数式を多く使ってきたと思いますが、これまでは主に数式は与えられて、それを計算したり、証明したりすることが多かったのではないのでしょうか。主体的にレポートを作成するにあたっては、**数式は「解く」ためのものである前に、数量的な関係を効果的に表現する手段**だという視点で図や表などの利用と同じように考えましょう。

数式を記述する際には使用されている変数などを正しく説明することが求められます。例えば

一定の速度で移動している物体の移動時間と移動距離の関係は

$$l = vt$$

であらわされる。

と書いただけではどの変数が何を表すか分かりません。次のように変数 l, v, t について本文中で説明する必要があります。

一定の速度で移動している物体の移動時間 t と移動距離 l の関係は

$$l = vt$$

であらわされる。ここで v は移動速度である。

13.3 変数は傾いている

数学など数式が書かれた教科書を見てみましょう。例えば2次関数について

$$f(x) = ax^2 + bx + c$$

と書かれていたりします。言われないと気づかないかもしれませんが、 x や y などの変数が x や y といったように**傾いた文字（イタリック体，斜体）**で書かれています。

これは斜体で書かれた文字は「**量を表す変数**」であること，すなわち，実際にはいろいろな値を取る得るものであることを示しています。数式には斜体で書くものと傾けずに立体（ローマン体）で書くものがあります。一般的には以下のようなルールに従っています。

- 量を表す変数は斜体で書く。
- \sin や \cos など定義の定まった関数や定数は傾けずに立体で書く。
- 単位を表す m や kg など傾けずに立体で書く。

13.4 物理量と単位の表記¹⁶

数学で扱う数式では数式は「数」を扱いますが，物理学などで扱う式では長さや質量などの「量」を扱い，具体的な量については 1.0 m のように，数値と単位の組み合わせで表します。

物理量を扱う場合は「数の式」ではなく「量の式」として扱い方が分かりやすくなります。例えば縦の長さ a ，横の長さ b の長方形の面積を S としたとき，それぞれの変数が「次元をもたない数」ではなく，「長さ」や「面積＝長さの2乗」の「次元」を持つ「量」を表し，量の式として

$$S = ab$$

と書くのです。具体的な物理量についても単位 m は「長さ」の次元を持つ量であると考え，

$$a = 2.0\text{ m}$$

$$b = 3.0\text{ m}$$

と書けばよく，さらに面積については

$$S = ab = 2.0\text{ m} \times 3.0\text{ m} = 6.0\text{ m}^2$$

と表記します。単位を「測定の基準となる量」として捉えることで両辺の次元が一致した記述ができます。こうすることで，単位の換算も例えば

$$m = 1000\text{ mm}$$

を上式の式に代入して以下のように計算できます。

¹⁶ この節で紹介している内容については文献[14]の6.2節や文献[15]でさらに詳しく議論されています。

$$S = 6.0 \text{ m}^2 = 6.0 (1000\text{mm})^2 = 6.0 \times 10^6 \text{mm}^2$$

13.5 Word での数式の書き方

Word で斜体や上付き文字，下付き文字を多用する数式を書くためには「数式ツール」を使います。

- 数式ツールの起動は以下のいずれかの方法で行います。
 - 「挿入」タブで「数式」を選ぶ。
 - Alt, Shift キーを押しながら = を押す。
- 変数など斜体にするには以下のように操作します。
 - Ctrl-i (Ctrl キーを押しながら i をおす) を入力する。

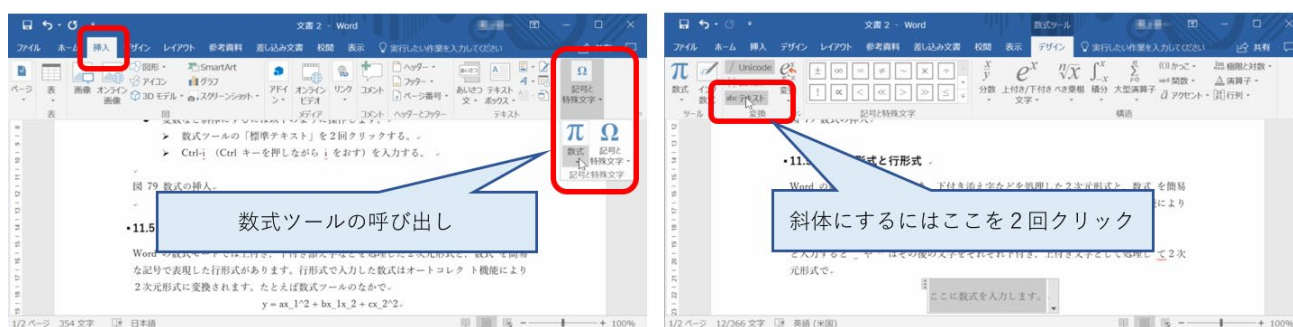


図 101 数式の挿入

13.6 2次元形式と行形式

Word の数式モードでは上付き，下付き添え字などを処理した 2 次元形式と，数式を簡易な記号で表現した行形式があります。1 次元形式に「unicode Math 形式」と「LaTeX 形式」が用意されています¹⁷。Web の記事などを参考にする際にはどちらの形式なのか留意してください。添え字などの多い数式を書く場合には 1 次元形式に慣れるとよいでしょう。

行形式で入力した数式はオートコorrect機能により 2 次元形式に変換されます。たとえば数式ツールのなかで

$$y = ax_1^2 + bx_1x_2 + cx_2^2$$

と入力すると $_$ や \wedge はその後の文字をそれぞれ下付き，上付き文字として処理し

¹⁷ 数式を含む文書を作成するツールとして LaTeX が有名ですが，その記法にそった記法が採用されています。

て 2 次元形式で

$$y = ax_1^2 + bx_1x_2 + cx_2^2$$

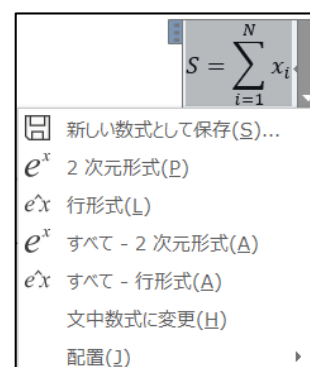
と表現されます。

13.7 文中数式と独立数式

数式では総和を表す Σ などが用いられますが、独立した行として書く数式では大きさや添え字の配置を読みやすくし、文の中での数式ではあまり上下に大きくせずに表現することが求められます。この表現を切り替えるのは「文中数式」と「独立数式」です。たとえば次の数式 $S = \sum_{i=1}^N x_i$ は文中数式として処理されていますが、これを独立した行では以下のように表現されます。

$$S = \sum_{i=1}^N x_i$$

左図のように数式のメニューの中で明示的にこれらの表現を切り替えることも可能です。



13.8 式番号と相互参照

本文中で数式を参照するには数式に式番号をつける必要があります。(Word 2016 以前のバージョンでは、数式ツールから式番号付数式を選ぶことで式番号を作成できます)

- 式番号付数式を作成し、式番号にラベルを設定します (図 101)。
 - 12.5 節の方法で数式ツールを使って数式を作成する。
 - 数式の最後に # を入力する。
 - 「参考資料」タブで「図表番号の挿入」を選ぶ。
 - 「オプション」の「ラベル」から「Equation」を選ぶ。
 - 「ラベルを図表番号から除外する」にチェックを入れる (番号を数値のみにするため)。
 - 「OK」をクリックすると#の後に番号が入力される (必要に応じ、番号の両側に半角括弧を入力する)。
 - 最後に数式の右端で ENTER キー を押す。

これで式番号が右寄せで表示されます。

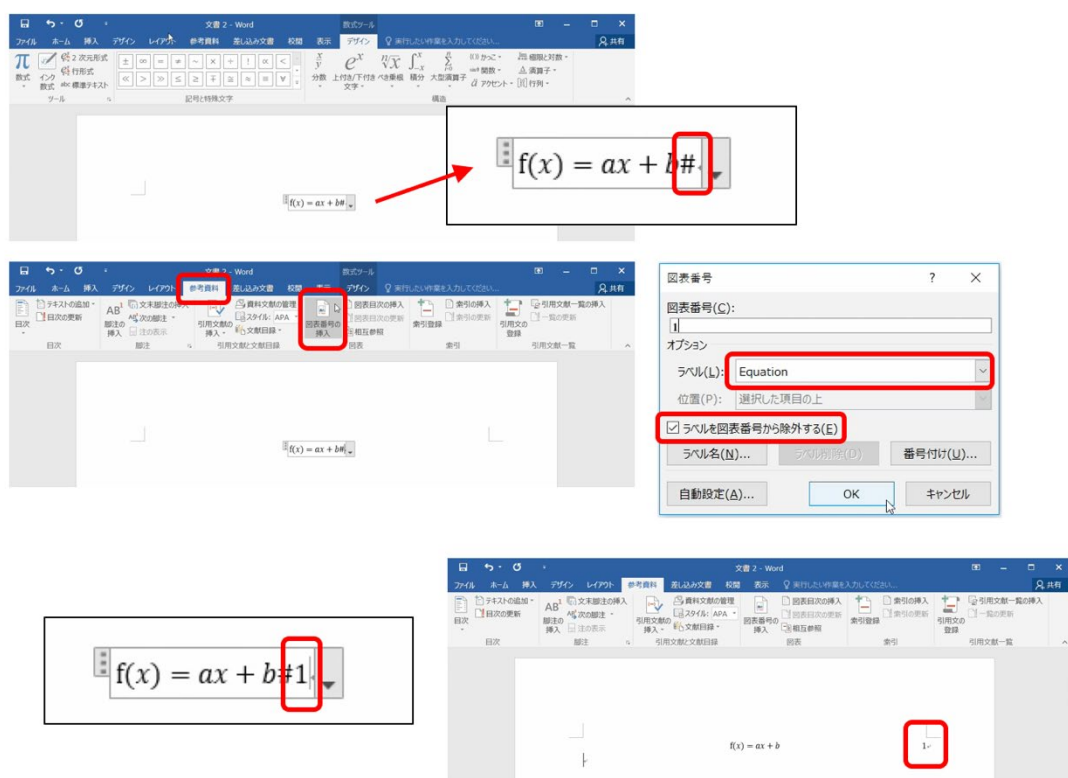


図 102 式番号付き数式の作成とラベルの設定

- 作成した数式に名前を付けて保存すると、以後、式番号付き数式の作成を簡略化できます（図 102）。
 - 式番号付き数式の右下に表示される ▼ から「新しい式として保存」を選ぶ。
 - 「新しい文書パーツの作成」ウィンドウで任意の名前（例えば「式番号付数式」）を入力し「OK」をクリックする（「ギャラリー」欄が「数式」であることを確認）。
 - 「挿入」タブで「数式」を選ぶと、保存した数式名が表示される。
 - この項目を選択すると、式番号が付与された数式を新たに作成できる（式番号は作成順に自動的に割り当てられる）。

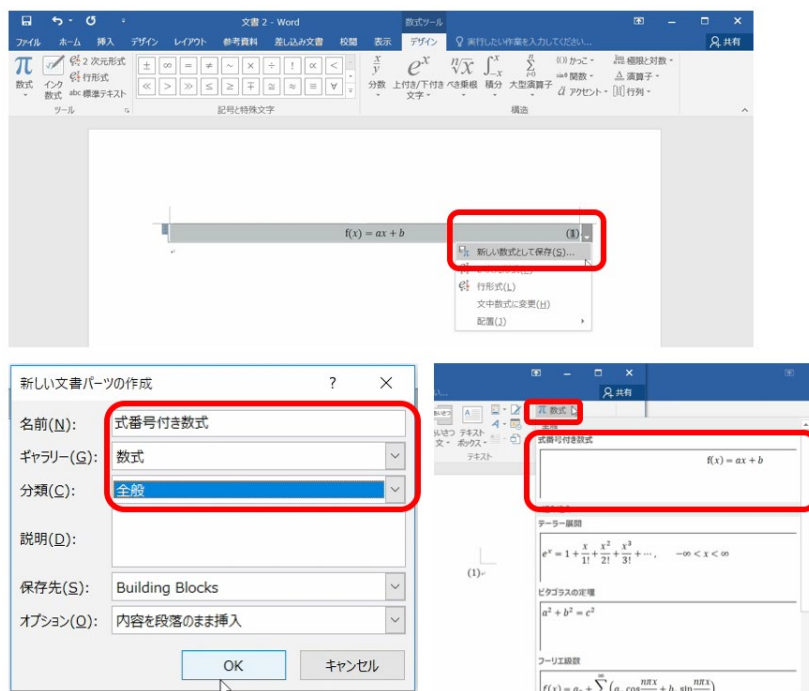


図 103 数式に名前を付けて保存

- 本文から数式を参照します（図 103）。
 - 参照したい数式の枠の中の式番号を選択した状態にする。
 - 「挿入」タブの「リンク」から「ブックマーク」を選ぶ。
 - 「ブックマーク」ウィンドウで任意の「ブックマーク名」を入力し「追加」をクリックする。
 - 本文の中からその数式を参照する場所にカーソルを移動し、「挿入」タブの「リンク」から「相互参照」を選ぶ。
 - 「相互参照」ウィンドウの、「参照する項目」は「ブックマーク」を、「相互参照の文字列」は「ブックマーク文字列」を選択する。
 - 「ハイパーリンクとして挿入する」にチェックを入れる。
 - 下側の「ブックマークの参照先」欄から、参照したい数式を選択する。
 - 「挿入」をクリックすると該当箇所に式番号が入力される。

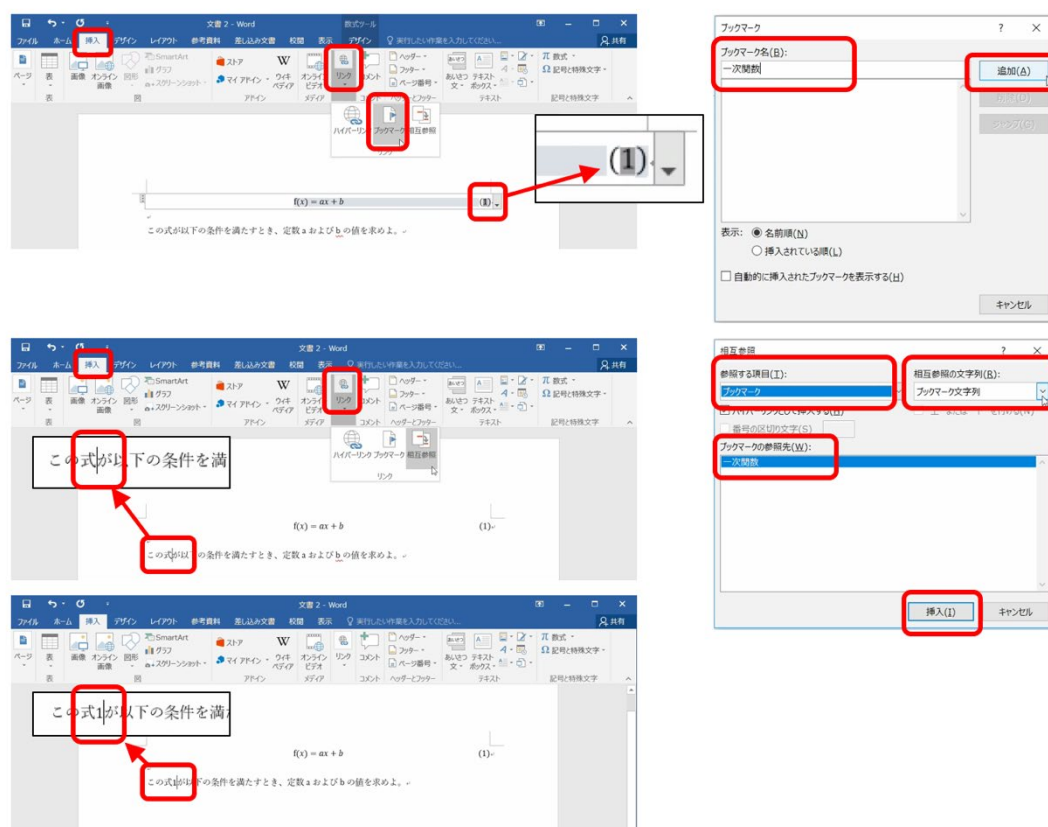


図 104 本文からの数式の参照

- Ctrl キーを押しながら本文の式番号をクリックすると、該当する式に移動します。

参考文献

- [14] 佐藤文隆，北野正雄：「新 SI 単位と電磁気学」，岩波書店（2018）
- [15] 森川鉄朗，西山保子：科学教育における量の計算法について，上越教育大学研究紀要，第 17 巻，第 1 号，pp. 365-375 (1997)

14. プレゼンテーションについて考えよう

14.1 学習の目的

- プレゼンテーションの目的や留意事項について学びます
- スライド作成の基本的な事項を学びます
- プレゼンテーションの作法について学びます
- 聞き手として望まれることを学びます

14.2 プレゼンテーションの目的と聞き手

演習 39. プレゼンテーションは何のために行うのでしょうか？

演習 40. プレゼンテーションに際して聞き手についてどのようなことを考えればいいのでしょうか。

- 聞き手はどのような人たちでしょうか
- 話し手が聞き手に期待することは何でしょうか

14.3 プレゼンテーションでの留意事項

14.3.1 時間を共有するということ

プレゼンテーションには与えられた時間があり、それは決して多くありません。プレゼンテーションでは聞き手に（強制的に）時間を使ってもらいます。多くの聞き手に時間を有効に使ってもらうことは発表者の責務です。

演習 41. 与えられた時間内に効果的にプレゼンテーションをするためには何をすればいいのでしょうか。

14.3.2 限られた時間で伝えられることの限界

プレゼンテーションやそのためのスライドは時間的な制約の中でコンパクトに情

報を伝えることに重きが置かれます。スライドはしっかり書かれた文章に比べ、情報量（文字数）が相当に少ないこと[16]には注意しなければなりません。複雑なことをしっかり伝えるためにはレポートなどの文章として表現してください。

14.4 スライド作成の基本

1) 枚数や文字の大きさの目安

スライドで使用する文字は読みやすい大きさを考えて決定する必要がありますが、**目安として10行程度**と考えます。ニュース解説などのテレビ番組でフリップが用いられるのを見ると相当に行数が少ないことが分かります。

プレゼンテーションに必要な時間の目安として、10行程度のスライドを作ると**スライド1枚を話すのに1分程度は必要**だと考えてください。もちろん、スライドに盛り込む内容の量によってその説明に要する時間は異なります、実際には練習などを通じて時間を見積もる必要がありますが、上記の見積もりで概略を想定します。

スライドのサイズは、初期設定では16:9ですが、プロジェクタが4:3のみに対応している場合があります。スライドのサイズを変更するには、「デザイン」タブで「スライドのサイズ」から利用するサイズ（「標準（4:3）」か「ワイド画面（16:9）」）を選択します。

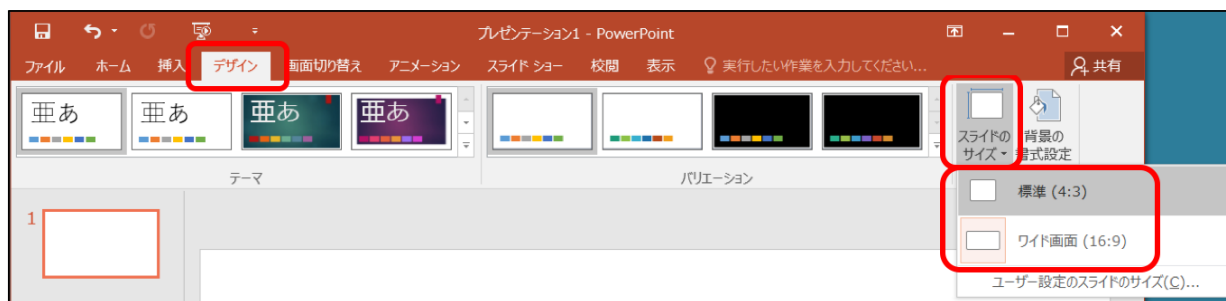


図 105 スライドサイズの変更

2) スライドの構成

発表時間を考慮してスライドに何を入れるかは以下の基準で考えてください。

1. 必要なものは入れて必ず話す
2. 不要なものは入れない
3. 論理を飛躍させない
4. 前から順に聞けば分かるようにする
5. 聞き手が何を理解できるか考える
6. 事実や他人の主張と自分（たち）の主張を明確に分ける

うまく話せないときには構成を見直します。

1. はプレゼンテーションを行う理由から明らかですが、限られた時間を聞き手に使ってもらうことから 2. も求められます。時間が限られていることから **1. については内容を重要なことに絞る**ことも必要になります。学術的な内容を扱う際は、3. や 6. は必須の要求です。また、内容が専門的になることが多いので 5. は聞き手への配慮として求められます。学術的な内容は、論理的な構成が重要で、なおかつ複雑な事項を話すことも多いのですが、「話す」ことが中心になりますので、内容を前後させない配慮として 4. が求められます。

3) 箇条書きについて考える

スライドは箇条書きを多用します。以下の点で考えましょう

- **同質なものが並んでいるか。** 箇条書きとする項目は並列して並ぶものですから、何らかの意味で同質なものが並んでいることが求められます。
- **順序に意味のあるのか、ないのか。** 作業の手順などは順序を変えると意味をなしません。このほか、重要度の順に並べるなど、順序に配慮することが求められます。
 - 順序に意味がなくても番号を振ると質疑をしやすいなど、よい点もあります。
 - 出典等を引用する場合は、もとの順序を守ることが求められます。発表者自身の論考のために順序を変える場合はそのことを明示する必要があります。
- **階層的に並べるべきものもある。** 例えば、この節で用いている箇条書きでも前の事項は順序に関する補足事項なので階層的に並べています。
- **箇条書きの項目数。** 箇条書きの項目数が少ないと、箇条書きにしている理由が分かりませんし、多すぎると理解が困難になります。発表者自身が項目数を決定している場合は次のことを考えます：
 - 項目数が少ないときには、考察が足りないことも多いので再考する。
 - 項目数が多すぎるときは、項目の整理が不十分なことも多いので再考する。

4) 表現の工夫

数値データはグラフ化する、数値そのものが重要な時に表にする。グラフのデータ点に数値を併記するという手法も有効です。

演習 42. 以下の2枚のスライドは同じデータを表現しています。その得失を考えてください。

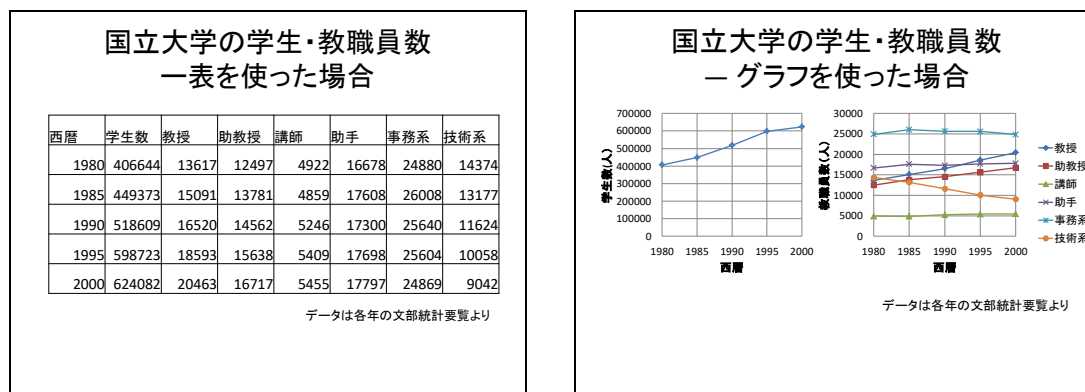


図 106 表とグラフの利用

スライドのなかでの強調

スライドで**一部を強調する**にはいくつかの方法があります。

- 文字の**大きさ**を変える
- 文字の**字体**を変える
- 文字の**色**を変える

多くの箇所を同時に強調することは難しいです。**最も重要な箇所だけを強調する**ようにします。

図 107 スライドのなかでの強調

- **強調** スライド中の文の一部を強調するには、
 - 図 106 に示すように**大きさ**、**字体**、**色**などを変えることで行います。
 - スライド全体で**統一的な強調方法**を使います。
 - 多くの箇所を強調すると相手に伝わりにくくなります。そのスライドで**最も重要なものを選んで強調**してください。
 - 単色のスライドの中で特定の所だけ別の色で文字を描くことは強調表現としては効果的ですが、**多くの色に意味を持たせて使うと、聞き手にとっては理解が困難**になります。

- **アニメーションの利用** スライドのアニメーションは見た目が面白いのでつい使いたくなってしまう。しかしながら、聞き手にとって本当に必要でなければ発表される内容に集中できないことも少なくありません。また、質疑などでスライドを行き来する際に手早く目的のスライドにたどり着けない、ということもあります。アニメーションは
 - 動的な内容を示すことや、
 - 相手に見せる内容を集中させたいときなど**真に必要な場所で使う**ようにします。
- **図や写真、ビデオの利用** 発表内容に具体性を持たせるのに効果的ですが、聞き手に与える効果が強いために不適切な図や写真の利用には注意すべきです。また、他者の権利に関わることが多いので以下の点を注意してください：
 - **著作者の権利への配慮**：イラストや写真を含め引用は著作権法に従ってください。スライドとして使用することと、それを印刷して配布すること、さらにはサーバ上にアップロードして他者が利用可能とすることは著作権法上の取り扱いが異なりますので留意が必要です。
 - **写真などに写っている人の権利への配慮**：人の肖像については「肖像権」として保護することが求められます。

14.4.2 パワーポイントのさまざまな機能の活用

1) アウトラインの活用

- スライドの構成にアウトライン（図 107）を活用します。
 - 文章として容易に書き込めます。
 - TAB キーや Shift + TAB キーでレベルを調整できます。
 - スライドの順番の入れ替えも容易です。

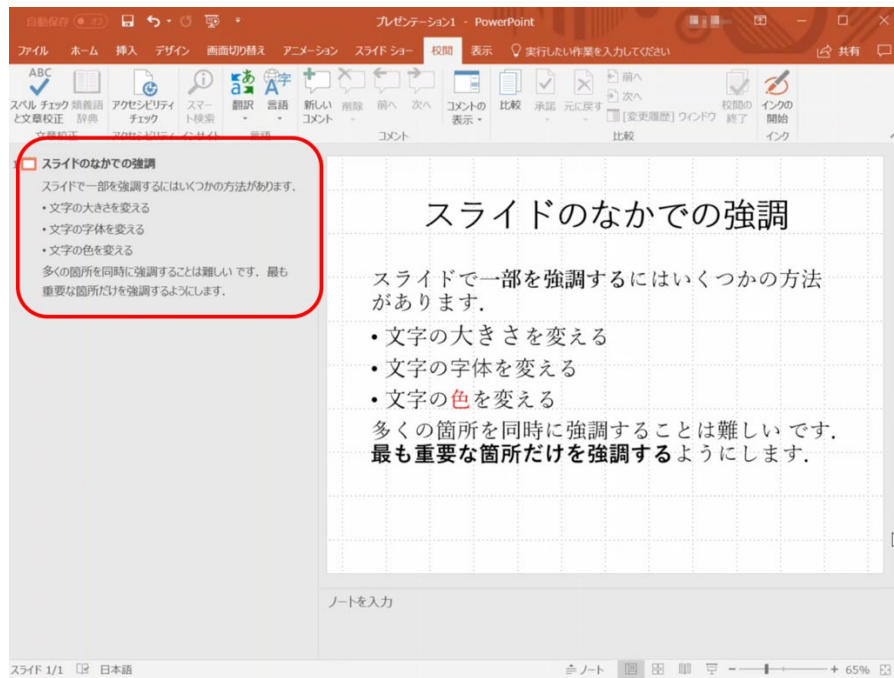


図 108 アウトラインの活用

2) スライド一覧の活用

- スライドの順番の入れ替えも容易です。
- 質疑などですばやくスライドを選ぶことができます。

3) スライドショーの開始とデュアルディスプレイの操作

- スライドショーを開始するには、画面下側の「スライドショー」のアイコンをクリックする方法と「スライドショー」タブの「スライドショーの開始」から選択する方法があります。後者は、現在表示しているスライドから再生するのか、最初のスライドから再生するのかを選びます。
- スクリーンにスライドを投影する際、PC側にスクリーンと同じ画面を表示させる方法と、前後のスライドやテキストのメモなどを同時に表示させる「発表者ツール」を使う方法があります。発表者ツールを使うには、「スライドショー」タブの「発表者ツールを使用する」にチェックを入れます。

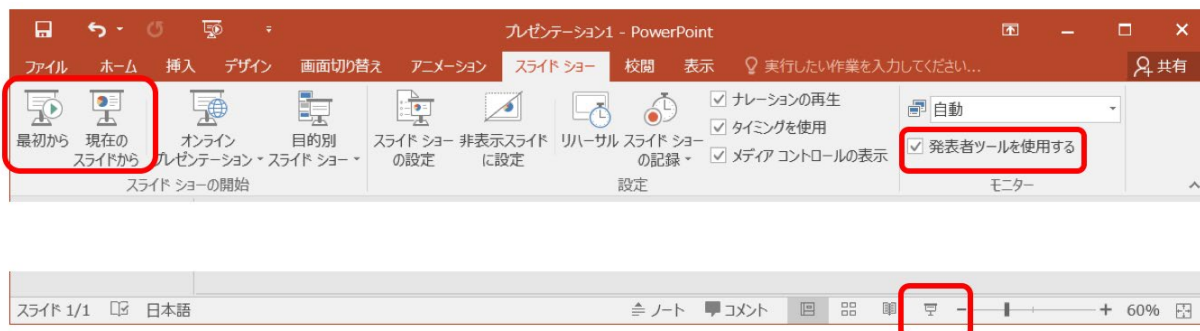


図 109 スライドショーの開始と発表者ツールの利用

4) スライド番号の追加

- 各スライドにスライド番号を入れておくと、質疑の際のやり取りをスムーズに行えます。
- スライド番号を入れるには、「挿入」タブで「スライド番号」を選び、表示された画面で「スライド番号」にチェックを入れます。「すべてに適用」をクリックすると、全スライドに番号が表示されます。

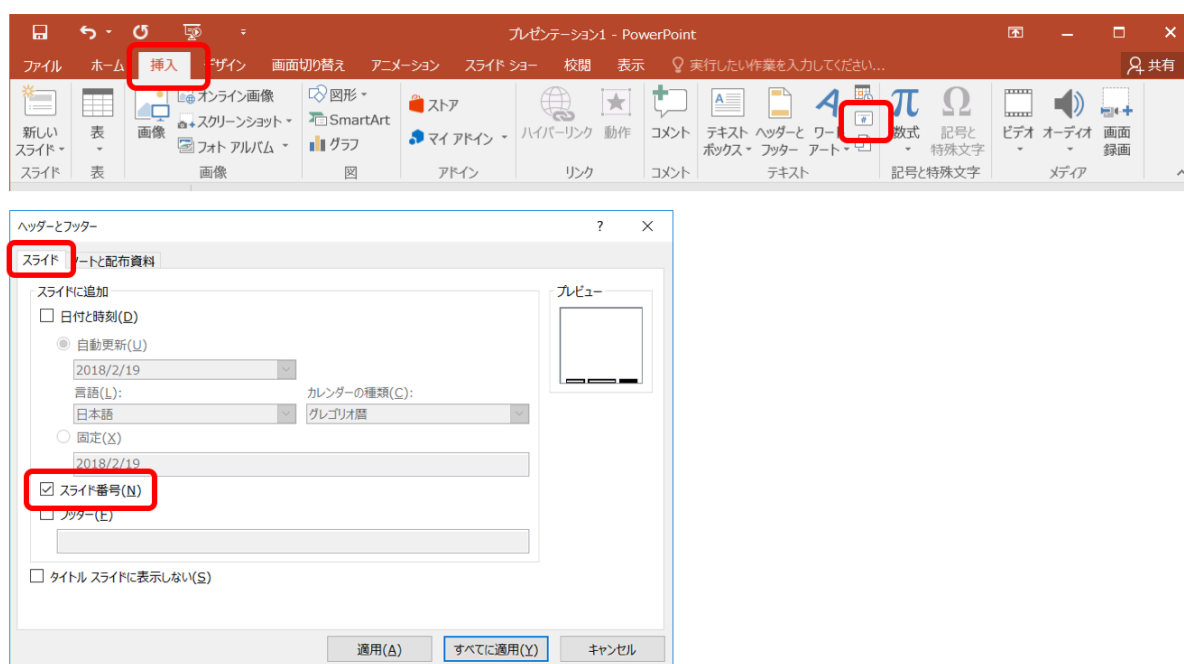


図 110 スライド番号の追加

5) スライドの印刷

- プリンタを使ってスライドの縮小刷りを行うには、「ファイル」タブで「印刷」を選びます。表示された画面上の「設定」で印刷レイアウトの中から1ページに表示するスライド数（2～6枚程度）を選択し、「印刷」をクリックするとプリン

タから出力されます。必要に応じてカラーなのかグレースケールなのかも選びます。

- 使用するプリンタによっては、印刷時のスライドのレイアウトを設定できることがあります。



図 111 スライドの印刷

14.5 発表の作法

1) 十分な準備

発表は**他人の時間を消費する行為であることを認識し**，十分な準備をしてください。

- **事前練習**：事前に練習，計時，改善します。**発表時間は厳守**します。
- **質疑への準備と対応**：質問を想定し，質疑に対する準備もしておきます。発表には含まなかった資料なども用意します。
- **発表に必要な機材**：P Cを持参するのか，発表会場で準備されるのかを確認します。
 - **プロジェクタとの接続**：P Cを持参する場合は，プロジェクタや大型ディスプレイへの接続方法が VGA なのか HDMI なのかを確認します。Mac など**アダプタ**が必要なものは忘れないようにします。
 - **音声の出力**：発表にビデオなどを含む場合，音声の再生をどのように行うかもチェックが必要です。多くの P Cには音声信号の出力用にステレオミニプラグの端子があります。大型ディスプレイなど投影機器が HDMI に対応している場合，音声信号も送信できることがあります。

- **ポインタ**：レーザーポインタや指し棒がつかえるのかどうかを確認します。
- **配布資料**：スライドの縮小刷りなど配布資料を準備する必要はないでしょうか。
- **筆記具**：質疑でのメモをとる用意をしましょう。
- **P C の操作への習熟**：ノート P C をプロジェクタに接続するテストは必ず事前に行います。留意点として以下が挙げられます。
 - **ノート P C の画面とプロジェクタ出力のモード**：ノート P C のディスプレイとプロジェクタに同じ画面を出すモードと、別の画面を出すモードがあります。このことを理解したうえで切り替えのための操作などを練習してください。
 - **プロジェクタの信号選択**：プロジェクタはさまざまな入力を選択できるようになっています。ノート P C からの出力を選んでいるかどうかを確認ください。
 - **プロジェクタとノート P C の相性**：ノート P C からの出力はさまざまな画面の解像度で行われます。通常、プロジェクタはこれに自動で対応しますが、場合によっては対応できないことがあります。ノート P C の画面出力の解像度を調整するなどで対応します。
- **トラブルへの備え**：機材がうまく動かない場合に備えてスライドのファイルを USB メモリにコピーして持参するなども有効です。

2) 聞き手に向き合ったプレゼン

手元やスクリーンばかりを見て話す方がおられますが、これではなかなか内容を相手に伝えることはできません。聞き手へのアイコンタクトをしながら、はっきりと聞こえる声で話すことが重要です。

3) 質疑での対応

質疑に適切に対応することは発表と同様に重要なことです。プレゼンテーションは内容を相手に理解してもらうだけでなく、聞き手の意見などを聞くことも目的としているからです。

- 質問、コメントはその場でメモを取るようにします。
- 回答する前に相手の質問をどう理解したかを確認することは良い方法です。
- 意味の分からない質問を受けることがあります。これに対しては自分なりの理解でよいか質問内容を確認するようにします。

- できていないこと、考えていなかったことは取り繕わずに正直に話しましょう。

14.6 グループでのプレゼンテーション

授業ではグループで調査などを行い、その成果を発表することがあります。

A,B,C,D さん 4 人が共同で発表することを想定してみましょう。

よくあるパターンは以下のようなものです：

- 自分が作ったスライド（1/4 ずつ）を入れ替わり 4 人で話す。

演習 43. 聞き手の立場にたってみて、これは効果的と言えるでしょうか？

演習 44. 4 人で共同してよいプレゼンをするほかの方法はあるでしょうか？

14.7 聞き手としての態度

プレゼンテーションについては聞き手としてのスキルを高めることも重要です。能動的に聞くためのスキルを身につけましょう。

- **批判的（否定的ではない）態度：**ほんとうにそうなのか、論理的なのか、証拠はあるのかなどの観点で発表を聞きます。
- **質問への準備：**批判的な態度で話を聞けば、確認したいこと、より深く理解したいことなど出てくるはずです。すべてを聞き終わったあとで質問を考えることは難しいので質問するつもりでメモをとるようにします。

14.8 プレゼンテーションの評価

もしプレゼンテーションを評価する、評価される、としたらどういう点を評価すべきでしょうか？以下のような評価の観点が考えられます。

- プレゼンテーションの内容
 - 興味ある、有用な内容であるかどうか
 - 聴衆を考えて理解できる内容であるかどうか
 - 内容が論理的に構成され、十分な論拠があるかどうか
 - 事実や他者の意見と発表者の意見を区別しているか
- プレゼンテーションの方法
 - 与えられた時間を適切に使っているか

- スライドの構成が適切かどうか
- しっかり話せているかどうか
- 図や表などを効果的に使っているか
- 視覚的効果は適切に使えているか
- 討論
 - 質疑に適切に答え、建設的な議論が行えているか

14.9 ポスター発表について

14.9.1 ポスター発表とは

研究などを発表する形式はスライドを用いて講演形式で行うもののほか、A0 や A1 サイズなどの大きな紙で作ったポスターを掲示し、その前に立ってポスターに書かれた内容について発表する形式も用いられます。講演形式の発表に比べ、並行して多くの発表を行うことができ、聴衆も興味のある発表を自由を選んで聴くことができます。

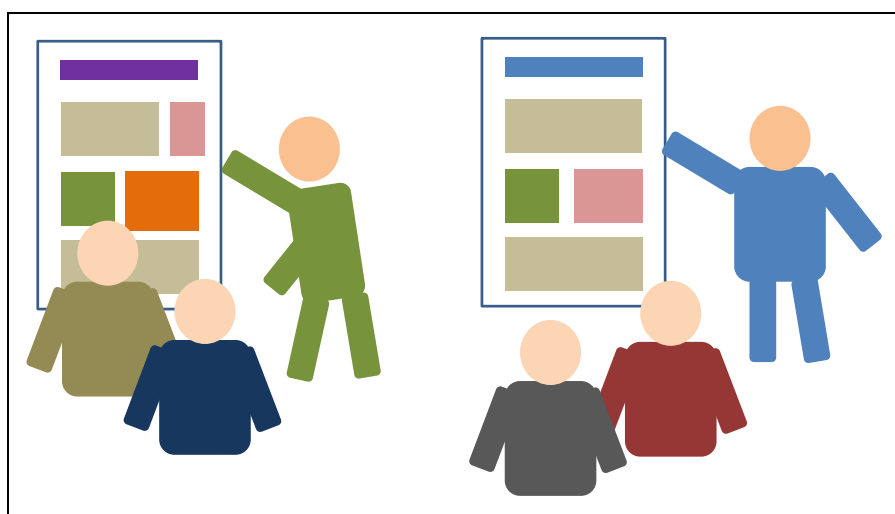
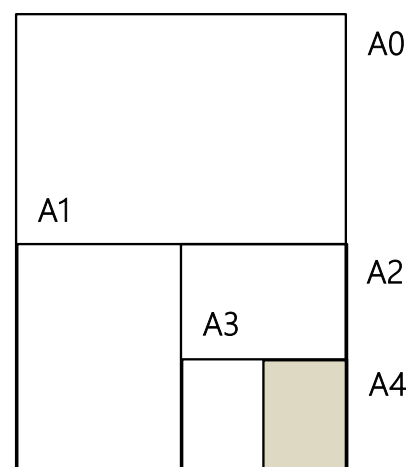


図 112 ポスター発表

14.9.2 ポスターの作成

ポスターの作成にあたっては以下のことに留意して作成します。

- ポスターを掲示する掲示板の大きさは発表会によって指定されていることが多いですが、A0、A1 などの大きさのポスターを用意することが多いです。A 系列は図 112 のように数字が 1 つ大きくなると半分の大きさになります。A1 だと A4 用紙の 8 枚分、A0 だと 16 枚分になります。
- パワーポイントでポスターを用意する場合は、ページのレイアウトで用紙サイズや向きを設定して作成します。
- 文字やグラフなどの大きさはポスターでの発表で聴衆の位置から読めるように設定します。
- ポスターは上部に発表のタイトルや発表者を分かりやすく表示します。
- 数分程度で発表することを想定して、文章やグラフなどを割り付けていきます。
- 大判プリンタでの印刷はかなりの費用が発生します。A4 や A3 サイズで印刷して、十分校正してから大きな紙に印刷するようにします。



14.9.3 ポスターでの発表

ポスターでの発表では以下の点に留意します。

- 数分程度で全体を発表できるように準備します。
- 発表会からの指示にしたがって掲示の方法（画鋏など掲示に必要なものを持参する必要があるのかどうか、など）を確認して準備します。
- ポスターを携行するための筒などを用意します。
- ポスター発表終了時の撤去方法も確認しておきます。
- より詳しく説明するための資料なども準備しておくといいいでしょう。
- ポスターの前でポスターを読んでいる聴衆には、「数分程度で発表しましょうか」と尋ねるようにします。
- 特定に人を相手に話すことが多いので、相手の理解を確認しながら発表します。

- 質問への回答などは講演形式での発表と同じで、メモをとったりしながら適切に対応します。

参考文献

スライドでのプレゼンテーションについてより詳しく知りたい場合は[17][18]を、ポスターについては[19]が参考になります。レイアウトやフォントの使い方については[20]が参考になります。

- [16] Edward Tufte: The Cognitive Style of PowerPoint: Pitching Out Corrupts Within, In Beautiful Evidence, Graphic Press LLC (2006)
- [17] 宮野公樹：学生・研究者のための使える！PowerPoint スライドデザイン 伝わるプレゼン 1つの原理と3つの技術，化学同人 (2009)
- [18] 宮野公樹：研究発表のためのスライドデザイン 「わかりやすいスライド」作りのルール，ブルーバックス B-1813，講談社 (2013)
- [19] 宮野公樹：学生・研究者のための伝わる！学会ポスターのデザイン術 ポスター発表を成功に導くプレゼン手法，化学同人 (2011)
- [20] 高橋佑磨，片山なつ：よい資料を作るためのレイアウトのルール 伝わるデザインの基本，増補改訂版，技術評論社 (2016)

15. プログラミングの基礎

ここではプログラムとプログラミングの基礎について学びます。演習に先立ってプログラミングの概要を把握するため、またさらに演習では触れないであろう発展的なトピックについて学習するためにぜひ活用してください。

15.1 学習の準備

プログラミングの学習では、資料を読むだけでなく、実践してみることが重要です。そこで本章の最後でプログラミングのためのツールの設定方法とその使い方をごく簡単に紹介します。

ところでプログラミングの演習に利用する道具立てはいろいろと考えられます。学習方法もさまざまでしょう。本章で紹介するのはそのうちの一つに過ぎません。

「情報基礎演習」のクラスは一つではなく、クラスによって演習で使うツールや演習の進め方は少なからず異なることでしょう。そこでみなさんの受講するクラスでは、本章で紹介するものとは異なるツールを利用する可能性も少なくありません。この点に注意して下さい。

15.2 学習の目的

「情報基礎演習」の仕上げとしてプログラミングを学習するのには次のような理由が挙げられます。

- プログラミングで（既存のツールでは対応できない）オーダーメイドの作業の自動化が可能になる
- プログラミングはコンピュータでの作業において同じことを何度も何度も繰り返す（退屈で）機械的な作業を避けるための手段となる
- プログラミングは面白い！（本当に！）

本来プログラミングの学習には時間をかけてじっくり取り組むのが望ましいところです。残念ながら本演習ではプログラミングの主要なトピックをすべてカバーするほどの時間はありません。しかし短い時間の中でもプログラミングの威力、面白さ（あるいは難しさ）を体験することはできるでしょう。この演習がプログラミン

グを本格的に学習するきっかけになることを期待しています。またプログラミングを学ぶことは、コンピュータをより深く理解するために役立つことでしょう。

15.3 プログラムとプログラミングの概要

最初にプログラムとプログラミングについて、まずそもそもそれが何なのかということから、プログラミングを学習することの意義と学習に対する心構え、またプログラムがどのようにコンピュータで使われるのかについて説明します。

- ソフトウェア（プログラム）とその役割
- プログラミングとプログラミング言語
- プログラミングの学習
- プログラム実行の仕組み
- プログラムに潜む危険性

15.3.1 ソフトウェア（プログラム）とその役割

コンピュータでさまざまなデータを処理することは、レシピを見ながら料理することになぞらえることができます。データ処理と調理を関係づけてみると次の表 12 のようになります。

- コンピュータによるデータ処理～レシピに従った調理
- ソフトウェア＝作業マニュアル
- さまざまなソフトウェア（マニュアル）→さまざまな処理

表 12 コンピュータでのデータ処理とレシピを使った調理

データ処理		調理
元データ	処理されるモノ	材料
ハードウェア	作業する主体	調理者
ソフトウェア (プログラム)	作業手順の情報	レシピ
結果のデータ	成果物	完成した料理

調理者はレシピの指示に従って料理の材料を順に加工（切る、まぜる、焼くなど）して料理を完成させます。調理者は淡々とレシピに従えば、たとえ料理を完成させるまでのイメージが頭に入ってなくても、悩む（考える）ことなく料理を作ること

ができます。レシピは（想定されるスキルがある人であれば）誰でも料理を作ることができるように書かれた調理マニュアルといえます。

さてすでに学習したようにコンピュータはハードウェアとソフトウェア（プログラム）で構成されています。

ハードウェアは実体としてのコンピュータを構成している機械です。調理でいえば作業をする調理者にあたります。ハードウェアにはデータの加工、保存、転送、つまりデータ処理を実現する多種多様な機能が備わっています。これらは調理者が持っているスキルに相当します。ただハードウェアがいろいろな機能をもっているとはいっても、機械であるハードウェアはそれだけで自律的に動作するものではありません。調理者もスキルをもっていたとしても料理の手順が頭に入っていないければ、材料から料理は作れないでしょう。

ハードウェアはいわば非常に複雑な電卓のようなものとも言えます。電卓には計算の機能は組み込まれています。しかし電卓が存在するだけで計算が独りでに進むことはありません。電卓で計算を行うには適切な順序で数値データを入力して演算機能を働かせる必要があります。人が数字や演算のボタンを押してその手順を逐一指示する必要があります。コンピュータのハードウェアでもデータ処理を実行するには、電卓と同様にさまざまな処理機能を適切な順序で駆動する必要があります。ただしコンピュータの場合、利用者が処理手順をその場で一つずつ順に指示しているわけではありません（もしそうだとすれば人間が操作するスピードでしかデータ処理が進みません！）。

コンピュータで高速なデータ処理を実現するためにハードウェアとともに不可欠な要素がソフトウェア（プログラム）です。調理において、レシピが材料から料理を完成させる手順をステップごとに並べたマニュアルといえるように、プログラムはデータ処理の手順を記述したコンピュータ用の作業マニュアルといえます。そこにはコンピュータがどんなデータを受け取って、どんな処理をどんな順序で施して、最後にどんな形で結果を示すかがまとめられています。

調理においては、（丁寧に書かれた）レシピが用意されていれば、調理者の頭に料理の完成までのイメージが入っていなくても、レシピの各ステップをこなしていくことで自然に料理ができあがります。プログラムに従ったハードウェアでのデータ処理もこれと似たようなものと考えられます。プログラムには（ハードウェアで実行可能な）データ処理のステップが順に記述されています。ハードウェアにはプログラムのステップを順に読み取って（コンピュータが考えることなく）その指示通りに動作するメカニズムが組み込まれています。データ処理の開始を告げるのは利用者の役目です（調理では調理者が自分のタイミングで自ら作業を開始するでしょ

う)。メニューの項目を選択したり、ボタンを押したりすることでコンピュータに対して処理の開始を指示します。そうすればあとはプログラム（マニュアル）の通りに（ハードウェアが動作するスピードで）処理が自動的に実行されるわけです。

調理において、一人の調理者が(持っているスキルの範囲で)いろいろなレシピ(調理マニュアル)に従っていろいろな料理を作れるように、コンピュータでは、(ハードウェアの機能の範囲で)さまざまなプログラム(作業マニュアル)に従って、さまざまなデータ処理ができるようになっていきます。なおレシピがあっても調理者にスキルがなければ料理が作れないように、プログラムを書いて実現できる処理はハードウェアに備わっている機能で実行できることに限られます。

15.3.2 プログラミングとプログラミング言語

ここではプログラムの作成(記述)に関する基本的な概念などについて説明します。

- プログラミング=プログラムの作成
- プログラミング言語=プログラム記述のための人工言語
- さまざまなプログラミング言語がある
- プログラムの記述には厳密性が求められる

1) プログラミングとは

さてソフトウェア（プログラム）はどうやって用意されるのでしょうか。コンピュータを購入すると、たいてい各種のソフトウェアが最初から組み込まれていて、（ちょっとした準備をすれば）すぐにコンピュータを使い始めることができます。

（設計済みの）ハードウェアの部品は人手によって作られるのではなく機械によって作られます。それではソフトウェアも機械で自動生成するのでしょうか。（残念ながら...??）そうではありません。ソフトウェア（プログラム）は人手で作られます。プログラムを作成することをプログラミングといいます。

2) プログラミング言語とは

すでに説明した通り、プログラムはマニュアル（手順書）のようなもので、プログラミングとはマニュアルを書くことを意味します。それではプログラムはどのように書くのでしょうか。処理の手順を日本語や英語で書くのでしょうか。

プログラミングには専用に作られた人工言語を用います。プログラミングに用いる言語を総称してプログラミング言語と呼びます。世界にさまざまな(人間の)言語が

あるように、プログラミング言語にもさまざまなものがあります。また（人間の）言語がその特徴によって分類できるようにプログラミング言語にもさまざまなタイプのものがあります。ここでメジャーなプログラミング言語をいくつか挙げてみます（表 13）。

表 13 プログラミング言語の例

C	C++	C#	Haskell
Java	JavaScript	Lisp	MATLAB
Perl	PHP	Python	R
Ruby	Scratch	VisualBasic	

なお情報科学，情報工学の分野では，人間の言語のことを（人工の）プログラミング言語と対比して自然言語といいます。

3) プログラミング言語における厳密性

自然言語と同様にプログラミング言語には（言語ごとに）文法が定められています。プログラムは文法に従って記述しなければなりません。そんなことは当然！と思うことでしょう。しかしちょっと待ってください。我々の自然言語を思い出してみてください。もちろん自然言語でも文法はルールです。しかしそのルールはある程度は寛容なもので、表現が多少曖昧であったり、間違いがあっても許される（理解できる）ことが少なくありません。一方で「プログラムを文法に従って記述する」というとき、それは文字通りルールは絶対であることを意味します。つまりプログラムの記述においてはたった一つの間違いも許されません。コンピュータが間違いを適切に解釈し直したり、行間を読んだりするようなことはまったく期待できません（そんな高等な機能は用意されていません！）。自分が書きやすいからといって、自分が書きたいようにプログラミング言語の文法を（ちょっとだけ）アレンジして書いてみても、それは全く受け入れられません（それぐらい分かるだろうって？、ブブー、残念でした）。ただしプログラミング言語には全く自由がないというわけではありません。空白の入れ方など、ある程度は自分で調整する余地もあります。しかしそれも文法で許容される範囲のことで、やはり文法を逸脱することは一切認められません。このようにプログラムにおいては厳密性が要求されます。

15.3.3 プログラミングの学習

ここではプログラミングの学習とは何を学ぶことか、プログラミングを学習することで何ができるようになるのか、またプログラミングの学習における心構えについて説明します。

- プログラミング→コンピュータを使いこなすスキル
- プログラミングの学習＝言語、データ表現、処理手順の学習

1) プログラミング言語の学習

プログラミングのスキルを身に付ければ、オーダーメイドの仕事プログラムで記述して、コンピュータに実行させることができます。プログラミングのスキルを習得するとは、いわばコンピュータを手足のように使いこなすスキルを身に付けることです。

プログラムを記述できるようになるためにはプログラミング言語を学ぶ必要があります。プログラミング言語は文法で規定されます。それでは文法を一通り学習すれば、すぐにプログラマ（プログラムを書く人）になれるのでしょうか。残念ながらそうではありません。プログラミング言語は「言語」です。英語を使いこなすために、まず英語の文法は習うとして、文法が分かったら英語が自由自在...なんてことがない（ふつうはそうですよね）のと同じで、プログラムを書くのにプログラミング言語の文法の学習は必要ですが、明らかにそれだけでは十分ではありません。

プログラミング言語を使いこなすには使い続けるしかありません（自然言語もそうですよね）。「習うより慣れろ」です。いろいろなプログラムを読んで書いてみることで、頭の中に言語のスキルが蓄積されていきます。そうすることで、はじめは教科書を片手にトットトとしか使えなかった言語が徐々に自分のものになって、そのうちストレスなく言語が操れるようになっていきます。

2) データ表現とデータ処理の方法の学習

さてプログラムを書くには、プログラミング言語を操れるだけでは不十分で、目的となる処理を実現する方法を見出すための素養が必要です。もう少し具体的には、処理するデータをどのように表現して、どういう手順でデータをどのように処理すれば目的を達成できるのかという問題解決の方法を発見する能力が必要です。このときとにかく目的が達成できれば（問題が解ければ）それで十分という場合もあるでしょう。しかし一般に同一の目的を達成するための処理の実現方法は一通りではありませんし、処理の方法によって効率が変わってくることもよくあります。効率のよいプログラムとそうでないプログラムでは処理時間に大きな差が生まれること

も少なくありません(問題を解くのに 10000 年かかるようでは困りますよね)。

データの表現方法やデータ処理の手順は自分で独自に考えることもできます。学習の過程では自分で工夫して、いろいろ試してみることも重要です。その一方で情報科学・情報工学の分野では、これまでにさまざまなデータ表現や処理方法が考案されてきています。プログラミングの学習においてはそれらを学ぶことも重要です。

以上のようなことから、プログラミングのスキルを（実用的な意味で）習得するには相当のコスト（学習時間）がかかることが分かるでしょう。しかし一度習得してしまえば、簡単に忘れてしまうことはありません。

3) プログラミングによる作業の効率化

PC やスマホなどのアプリケーションはすべてプログラムです。ネット上のサービスもプログラムによって実現されています。プログラミングのスキルを身に付けることで、自分でも独自に新たなアプリケーション、サービスなどを実現できるようになります。

それではそのような開発に携わらない限りプログラミングは役立たないかといえ、そうではありません。プログラミングは新たなツールを作るだけでなく、コンピュータでの作業を効率化するのにも役立ちます。コンピュータで同じような作業を延々と繰り返すことになるとき、プログラミングのスキルをもっていれば、プログラムを書いた方がよいと考えることもできるでしょう。たとえば「File00.txt」

「File01.txt」...「File99.txt」という 100 個のファイルの名前を「File000.txt」「File001.txt」...「File099.txt」に変えたいとします。これを一つ一つ手作業でこなすことは不可能ではありません。しかしその作業は退屈な上に面倒で間違いやすく時間もかかることでしょう。このときちょっとしたプログラムを書けば、同じ処理を一瞬で終わらせることができます。プログラミングのスキルはコンピュータを扱う上で大きな力になりえます。

4) 失敗を恐れない

さて誰しも新しいことに挑戦するときには失敗をするものです。プログラミングの学習過程でも失敗はつきものでしょう¹⁸。作成したプログラムに文法的な間違いがある場合はプログラムを実行（の準備を）しようとしたときに自動的に検出されます（そのときに出力される「エラーメッセージ」をヒントにして誤りを修正することになります）。プログラムが文法上正しくてもうまく動作しない場合もあります。しかしそのようなプログラムを実行したとしてもコンピュータが壊れることはまず

¹⁸ 十分に学習した後でも、きっと失敗はなくならないでしょう...;-P

ありません¹⁹。学習段階では大いに失敗しても構いません。むしろ「失敗は成功のもと」と言われる通り、失敗から多くのことが学べるでしょう。

15.3.4 プログラム実行の仕組み — プログラミング言語とハードウェアの狭間にある機械語

ここではプログラミング言語で記述したプログラムに従ってコンピュータがデータ処理を実行する仕組みについて説明します。このトピックは発展的なもので少しとっつきにくい内容かもしれませんが(おそらく授業では扱われないでしょう)。またここで説明することを知らなくてもプログラムを作成して動作させることはできます。しかしそのような仕組みを理解しておけば、プログラミングの学習で発生するであろう「なんでそうなの?」「なんでそうするの?」といったさまざまな疑問を解消するのに役立つでしょう。

- プログラミング言語＝人間のための言語
- プログラミング言語のプログラム→機械語に翻訳して実行

1) プログラミング言語による記述とハードウェアによる処理とのギャップ

さて次に Python というプログラミング言語で記述した（ごく短い）プログラムの例を示します。

リスト 1 Python プログラムの例

```
n = int(input("n? "))

fact = 1
for j in range(1,n+1):
    fact = fact * j

print('{0}!={1}'.format(n, fact))
```

Python を知らなければ、このプログラムを理解することは難しいでしょう（いまこのプログラムを理解する必要は全くありません）。しかしそれでも、このプログラムには英単語や数式らしきものが見られますし、Python について学習すれば読んで

¹⁹ 機械を制御するプログラムに間違いがある場合は機械を壊してしまう危険性があります。しかし学習段階ではそのようなプログラムに触れることは少ないでしょう。

理解することは期待できそうです（そう思えますよね?）。

さてプログラムはプログラミング言語で記述します（Python であれ何であれ）。コンピュータは与えられたプログラムの指示に従ってデータを処理します。それでは具体的にコンピュータはプログラムをどのように解釈して指示を実行しているのでしょうか。

コンピュータでデータ処理を行う実体はハードウェアであることを思い出してください。ハードウェアは単なる機械で要するに電子回路の集積物です。それではそのような機械がどのように言語の記述に従って処理を実行するのでしょうか。プログラミング言語を「賢い」コンピュータが学習するのでしょうか（一体どうやって??）。

2) 機械語 — ハードウェアを駆動する言語

プログラミング言語で記述されたプログラムと機械であるハードウェアの間には明らかに大きなギャップが存在します。このギャップを埋めるのが機械語です。プログラミング言語で記述されたプログラムはそのままハードウェアで解釈されるものではありません。ハードウェアで行われている処理とは、突き詰めればコンピュータ内のさまざまな装置に格納されているさまざまなデータの電子回路による加工や転送です。電子回路はメモリ装置に格納されたデータ（物理的な状態で表現されている 0,1 のビット列）を処理装置に適宜送り込むことで駆動されます。0,1 のデータによって、さまざまな回路のスイッチが ON/OFF されるわけです。どんなビット列のデータがハードウェアのどの回路をどう駆動するのかは機械的な仕組みとして定められています（そのようにハードウェアが組み立てられています）。

さまざまな「ハードウェアでの処理を駆動するビット列」によって定められるのが「機械語」です。例えばメモリのある場所に別の場所からデータを送り込む処理を駆動する機械語の命令（ビット列）やあるレジスタ（特殊なメモリ）のデータを別のレジスタのデータに加算する処理を駆動する機械語の命令（ビット列）などが定められています。そのような機械語の命令（ビット列）を適宜並べることでハードウェアを駆動する「本物」のプログラムである機械語のプログラムが得られます。

次に機械語プログラムの（ほんの）一部を例として示します。これは決してデタラメに作ったものではなく、本物のプログラムから抜粋したものです。機械語のプログラムのビット列にはもちろん意味が定められています。しかし簡単に読み書きできる代物ではありません。これを見れば、プログラミング言語で書かれたプログラムの方が遥かに分かりやすそうなのが実感できますよね。

```
0000000001101100011010010110001001100011001011100111001101101111
0010111000110110000000000111001101100101011101000111010101101001
011001000000000001100101011110000110100101110100000000001100101
0111100001100101011000110111011000000000011001100110111101110000
0110010101101110000000000111001101110100011100100110111001100011
0110110101110000000000000111000001100101011100100111001001101111
0111001000000000010111110101111101101001011100110110111101100011
0011100100111001010111110111001101110011011000110110000101101110
```

図 114 機械語のイメージ

3) プログラミング言語と機械語のギャップを埋める仕組み

それでは機械語のプログラムと Python のようなプログラミング言語のプログラムはどのような関係にあるのでしょうか。プログラミング言語で記述したプログラムはどうやって解釈されて実行されるのでしょうか。

コンピュータが開発された当初、プログラミング言語は存在せず、すべてのプログラムは機械語で書かれていました。機械語によるプログラミングにおいては、ハードウェアでの処理を具体的に指示する機械語の命令を順に並べてデータ処理の全工程を記述することが求められます。ハードウェアの内部構造を知らなければ機械語プログラムは書けません。またハードウェアで実行される処理の一つ一つはごく単純であるため、機械語のプログラムは長くなりがちで意味を把握するのが難しくなります。さらに機械語命令はハードウェアの回路を駆動するための 0,1 のビットの並びでしかなく、単語のようには読めません。このように機械語のプログラムは読み書きに大変な苦勞を伴います。また機械語の記述はハードウェアの動作に直結しているため、ハードウェアが異なれば、当然機械語も別のものになります。つまりあるコンピュータ用に書いた機械語プログラムはハードウェアが異なる別のコンピュータでは動作しません。

以上のように機械語によるプログラミングは決して効率的ではありません。そこでまずプログラムを楽に読み書きできるようにするために、英単語や数字で機械語を直訳して記述できるようにしたアセンブリ言語がコンピュータが現れてから早い段階で開発されました。それと同時に「アセンブリ言語を機械語に翻訳する」機械語プログラムであるアセンブラも開発されました。

アセンブリ言語は機械語より遥かに扱い易いものですが、それでも機械語を直訳

ただで機械語の欠点をすべて解消できたとは言えません。そこでハードウェアに依存せず、さらに読み書きしやすいプログラミング言語および「プログラミング言語で記述したプログラムを機械語に翻訳する」プログラム²⁰が開発されました。ちなみに世界初のプログラミング言語は FORTRAN で 1957 年に発表されています²¹。

以上のようにコンピュータが実行しているのは（ハードウェアで機械的に解釈できる）機械語のプログラムで、プログラミング言語で書かれたプログラムは機械語に翻訳されて実行されるわけです。

Great Idea! 自動プログラミング

プログラミング言語で書かれたプログラムを機械語に翻訳する作業自身をコンピュータにさせることは「自動プログラミング」とも呼ばれます。「プログラムを作ること」自身を「プログラム」で行わせることで、複雑なプログラムの設計・開発・保守が可能になりました。

15.3.5 プログラムに潜む危険性

ここではプログラムの誤りによって引き起こされる問題について説明します。さてコンピュータはプログラムで指示されたことは忠実にこなします。そこで間違ったことを指示したとしてもそのままその通りに実行します。またどんなデータが与えられてもプログラムで受け入れ可能であれば処理します。コンピュータは「おっと、このプログラムは何だかおかしいぞ...」「このデータは怪しいな...」などと考えたりはしません。

プログラムの中に間違いがあるとき、それを実行すると処理の途中でプログラムが異常終了(abort)することがあります。逆に処理が延々とつづいて終わらなくなってしまうこともあります。別の問題として、処理は完了しても、その処理方法が間違っている場合があります。そのことにより誤った結論が導かれ実害が発生することもあります。たとえば人工衛星に搭載されたプログラムで数値データの正負を誤って記述したため、コントロールがおかしくなって、衛星を破棄せざるを得なくなったという問題が知られています²²。

²⁰翻訳プログラムには大きく分けてコンパイラとインタプリタがあります。コンパイラでは機械語のプログラムを生成します。我々は生成されたプログラムを実行することになります。一方インタプリタの場合はプログラミング言語のプログラムを読み取りつつインタプリタに組み込まれた機械語の命令を実行します(機械語のプログラムは生成しません)。なおプログラムの実行スタイルは他にもあります。

²¹ IBM100 FORTRAN The Pioneering Programming Language
<http://www-03.ibm.com/ibm/history/ibm100/us/en/icons/fortran/> (Last access: 2016-08-29)

²² X線天文衛星「ひとみ」の異常に関する小委員会の検証結果について
http://www.jaxa.jp/press/2016/06/20160614_hitomi_j.html (Last access:2016-08-29)

またさらにプログラムで想定されている（通常の）利用方法において全く問題がなかったとしても、プログラムに（見落とされている）間違い、つまり（隠れた）欠陥がある場合、想定外のパターンのデータをプログラムが受け付けてしまうことで、問題が発生する場合があります。マルウェア（コンピュータウイルスなど悪意の目的で使われるソフトウェア）は、そのような欠陥を悪用しています。

プログラムに含まれる間違いを専門用語でバグ（bug）とよび、プログラムを修正してバグを取り除くことをデバグ（debug）といいます。セキュリティ上の問題となる欠陥（バグ）は特に脆弱性（vulnerability）とよばれます。

さまざまなシステムがコンピュータ制御されている現在の社会において安全・安心に生活をするためには、バグのないプログラムを作ることが求められます。しかしプログラムを作るのはあくまでも（間違いを犯すことのある）人間です。一般に大規模で複雑なプログラムになるほどバグを完全になくすのは困難になります。

15.4 変数によるモデル化

ここではプログラムの不可欠な構成要素として、変数について説明します。変数とは何か、どう使われるのか、また変数を使うことの意義について説明します。

15.4.1 例題 — コラッツの予想

さてここで例題として、数学でよく知られている問題である「コラッツの予想」（Collatz Problem）を紹介します。

コラッツの予想

任意の正の整数 n から始めて、偶数なら 2 で割る、奇数なら 3 倍して 1 を加えることを繰り返すといつかは必ず 1 になる。

たとえば $n=3$ についてコラッツの予想が正しいことは次のように確かめられます。

$$3 \rightarrow 10 \rightarrow 5 \rightarrow 16 \rightarrow 8 \rightarrow 4 \rightarrow 2 \rightarrow 1$$

この問題は一見ごく単純ですし、実際にいろいろな数で試してみても予想は正しように思えます。しかし数学的には正しいことが証明できていません。つまりコラッツの予想は未解決の問題です。

数学の証明問題としてコラッツの予想に取り組むことは至難の業でしょう。しかし具体的に整数 n を指定して、予想が成り立つかどうか試してみることはできます。

次にそのようなテストを実行してみるプログラムをプログラミング言語 Python で示します。

リスト 2 コラッツの予想を試すプログラム

行番号	プログラム	解説
1	<code>n = int(input("n? "))</code>	キーボードから <code>n</code> の値を得る
2	<code>while n != 1:</code>	<code>n ≠ 1</code> である限り 2～7 行目の処理を繰り返す
3	<code> print(n,end = " -> ")</code>	途中経過の表示(現在の <code>n</code> の値)
4	<code> if n % 2 == 0:</code>	<code>n</code> が偶数(2 で割って余りが 0)のときは
5	<code> n = n // 2</code>	→ <code>n</code> を 2 で割った値を再び <code>n</code> とする
6	<code> else:</code>	そうでないとき(<code>n</code> が奇数のとき)は
7	<code> n = 3*n+1</code>	→ <code>n</code> を 3 倍して 1 を加えた値を再び <code>n</code> とする
8	<code>print("1 OK")</code>	ここに至った時点で <code>n=1</code> のはず

このプログラムでは `n` として正の整数が指定されることを期待して、その `n` について予想が正しいかどうかを具体的に確かめます。計算過程での `n` の値の変化を順に表示していった、予想が実際に正しいのであれば最後に「OK」と表示してプログラムは終了します²³。

以下ではこのプログラムでの変数の必要性を説明します。また一般に変数を利用する意義を説明していきます。

15.4.2 変数とその必要性

- プログラムのデータは主に変数で扱う
- 変数＝データと名前を対応づける仕組み、データのモデル

さて例題のプログラムはさまざまな `n` の値についてコラッツの予想が正しいかどうかをテストできるようになっています。プログラムでは `n` の値は指定されていません。つまりプログラムの記述においては `n` の値は不定で、実行するときに初めて `n` の値が定まるようになっています。また `n` が偶数なら 2 で割る、奇数なら 3 倍して 1 を加えるという計算の結果得られる値を再び `n` に指定して処理を行っています。

このようにさまざまな値と結び付けられるデータは、プログラムにおいてはその

²³ `n > 1` からスタートして 1 に辿り着く前に同じ数が 2 回出現したら、コラッツの予想が間違っていることが分かります。しかしこのプログラムではそのことを確かめるようにはしていません。プログラムの中の「//」は Python で商を計算する演算子です。

データを象徴する名前によって表現されます。またプログラムが実行されるときには名前にデータの値を当てはめて処理が行われます。名前とデータを対応付けて扱う仕組みを変数といいます。これは数学で扱う変数と同様のもので、プログラムでデータを表すモデルであるとみなせます。

15.4.3 変数の意義

- プログラム実行における変数の意義
 - プログラムの外部のデータの処理
 - プログラムの汎用化
 - 計算結果の再利用
- プログラムを書くときの変数の意義
 - 名前でデータに意味づけ
 - データとデータの関係の明示
 - プログラムの修正を容易にする

1) プログラムの実行における変数の意義

プログラムで変数を使う理由の一つとして、すでにコラッツの予想の例で見たように、プログラムの外部からデータを取り込んで処理することが挙げられます（そのようなデータを予めプログラムに記述しておくことはできません）。

また変数にはさまざまな値を当てはめられるため、同一のプログラムで（同種の）さまざまな値に対する処理の結果を得ることができます。つまりプログラムに汎用性をもたせることができます。数学の関数と同じで、変数を使うことで処理をモデル化（計算方法を名前を使って一般的に表現）できるわけです。

変数を使えば処理の効率を上げることも可能です。いま仮に、ある関数 $f(x)$ について $f(10000)$ の計算に 100 秒かかるとしましょう。このとき次の計算を考えます。

$$f(10000) \times f(10000) \times f(10000)$$

この処理では $f(10000)$ を 3 回計算するため、計算には全体でおよそ 300 秒かかることが予想されます（乗算はたいていほぼ一瞬で終わります）。このとき $f(10000)$ は同じ値に決まっているからといって、2 回目以降の計算が（自動的に）省略されたりはしません。コンピュータはプログラムに指示された通りにしか処理をしません。ここで $f(10000)$ とは $x=10000$ のときの $f(x)$ の値を計算することを意味しています。プロ

グラムにそう書かれている以上、書かれている通りに計算処理が実行されます²⁴。

さてここで次の式によって数学のように変数 y に $f(10000)$ の値を代入できるものとします。

$$y = f(10000)$$

ここでさきほどと同じ結果になるような計算処理を次のように記述したとします。

$$\begin{aligned} y &= f(10000) \\ y \times y \times y \end{aligned}$$

このように記述したとしても計算結果はさきほどと全く変わりません。しかしこの場合、 $f(10000)$ を計算するのは 1 回だけです。そこで処理時間はほぼ 100 秒に短縮できるでしょう。このように変数に途中結果を代入して再利用することで処理を効率化できます。なおプログラムにおいて同じ結果が得られるデータ処理を何度も実行することは無駄と考えられます（処理時間が大きく変わることがあります）。

2) プログラムを書くときの変数の意義

さてプログラムにおける変数の威力は、実行時のみに発揮されるわけではありません。プログラミングにおいても変数は必須です。

データの値は無個性です。たとえば「10」という数値があったとして、その数値だけを見ても、それが何のデータなのかは分かりません。「10 番」かもしれませんし、「10 秒」かもしれませんし、「10m」かもしれません。また同じ値があちこちにあったとしたら、それらは区別できませんし、たとえ同じ値であったとしても、それらが同一の情報を表すデータとは限りません。

それに対して変数によってデータを表現すれば、その名前でデータに意味づけます。また変数を組み合わせて処理を記述することで、どんなデータからどんなデータが得られるのか、データとデータの間を明示することもできます。たとえば次の(a),(b)を比べてみてください。print は画面にデータを表示する機能だとします。また「*」は乗算、「/」は除算を表しています。

リスト 3 値のみのプログラムと変数を利用したプログラム

(a)	print(100 * 12 / 2)
-----	---------------------

²⁴賢い翻訳プログラムならこのような重複した処理を検出して重複を除くように最適化するかもしれません。

(b)	<pre>base = 100 height = 12 area = base * height / 2 print(area)</pre>
-----	--

(a),(b)とも実行すれば同じ値が得られます。(a)の方が記述は簡単です。しかし見ただけでは一体何を計算しているのか分かりにくくなっています。一方(b)であれば、処理の意味はわかりやすいのではないのでしょうか。

なおプログラムにおいて、値が直接記述されていて意味が分かりにくいデータのことをマジックナンバーといいます。マジックナンバーは避けるべきです（自分が書いたプログラムでも後で見たときには意味が分からなくなってしまうかも！）。

変数を使えば、プログラムの修正も容易になります。同じデータの値をあちこちにそのまま書いたとすると、値の変更が必要になったときには、それらをすべて適切に置き換えることになります（おっと、たまたま同じ値になっている別のデータを書き換えないように！）。データを変数で表現しておけば、変数に代入する値を置き換えるだけで作業は終わりです。

以上のように、プログラム、プログラミングにおいて、変数は欠かすことができない仕組みです。

15.5 制御構造 — 条件に基づく処理

ここでは変数とともにプログラムで本質的に必要となる制御構造について説明します。制御構造とは条件に基づいた処理の仕組みの総称です。制御構造には一般に「条件分岐」と「繰り返し処理」があります。とくに「繰り返し処理」によって大量のデータを順に一つずつ処理するために何度も同じ（パターン）の処理を繰り返すことはコンピュータの真骨頂と言えます。

- 条件に応じて異なる処理を行う
- 条件に従って同じ（パターンの）処理を繰り返す
- 繰り返し処理→コンピュータの真価の発揮
- 停まらないプログラムもある

15.5.1 条件に基づく処理の必要性

先に挙げた「コラッツの予想」をテストするプログラムでは条件を考慮した処理

が必要となります。具体的には、まず n が偶数か奇数かによって計算処理が異なります。また n が 1 と等しくなるまで同じ処理を繰り返します。これも「 n が 1 に等しくなるまで」という条件に従った処理です。プログラムではこのように条件に基づく処理が必須です。

さてプログラムにおいては命令が一つずつ順に処理されていくのが基本動作になります。これはマニュアルに従った作業で記述されているステップを最初から最後まで順番に辿るのと同じようなものです。一方で条件に基づく処理では、条件によって異なる処理が実行されたり（条件によっては実行されない処理が出てきます、これはマニュアルでもよくあることです）、条件に基づいて同じ処理が何度も繰り返されたりする（プログラムの同じ範囲の処理を繰り返します）ことになります。このようにプログラムの処理の実行順序が条件によってコントロールされる仕組みを一般に制御構造といいます。

15.5.2 繰り返し処理の威力

さてコンピュータを使ってデータを処理することの大きなメリットは、何千、何万、何百万という（一連の）大量のデータの処理、あるいは延々と続く計算処理を高速にこなせることです。これはプログラムに繰り返しの制御構造を組み込むことで実現されます。同じ作業を延々と繰り返すことは人間には苦痛ですし、繰り返すうちに集中力を欠いて間違えることもあるでしょう。しかしコンピュータにはそのような問題はありません。与えられたデータがいくら多くても、疲れることなくいくらでも同じ処理を忠実に繰り返すことができます（電力が供給されてハードウェアが動き続ける限り）。繰り返しの制御構造がこの力の源となります。

たとえば次の数を n として「コラッツの予想」をテストするプログラムを実行すると 2813 回の繰り返し後に 1 に到達することが分かります²⁵。

5293237546378456534784553489308590438902375927369027359027902375429185097690729571298679027590211819

このことを手で計算して確かめるのはまず不可能ですが、コンピュータに計算させればにすぐに結果が得られます。これこそがコンピュータの力です。

²⁵ここで 1 に到達するまでの処理回数を知るにはプログラムに繰り返しの回数を記憶させる処理を追加する必要があります。なおプログラミング言語によっては、ここで示したような「大きな」数と通常の計算で扱うような「小さな」数では異なる処理が必要になります。これはコンピュータ内でのデータの表現が「小さな」数と「大きな」数では異なるためです。

15.5.3 停止しないプログラム

さて繰り返し処理についてもう少し説明します。これまでに見てきたように繰り返し処理とは「条件が正しい限り繰り返す」処理です。ここで「ファイルの各行を順に読み取ってそれぞれに同じ処理を施す」ような処理を考えてみましょう。これは「ファイルから行が読み取れる」という条件による繰り返しで実現できます。具体的にはファイルの先頭から1行ずつ読み取って処理することを繰り返すと、いつかは最後の行に到達して、その次に「ファイルから行が読み取れる」という条件が成り立たなくなってしまう。このような処理は（ファイルがいくら長くても）いつかは終わります。

それでは全ての繰り返し処理は「いつかは終わる」ことが決まっているのでしょうか。このような問いかけがなされることから想像できるように、ずっと終わらない繰り返し処理もありえます。単純には繰り返す条件として「常に正しいこと」を指定すれば、（コンピュータが動作する限り）永遠に終わらないプログラムになることが分かるでしょう。次にそのようなプログラムの簡単な例を示します（とくに意味のあるものではありません）。

リスト 4 停まらないプログラムの例

行番号	プログラム	解説
1	<code>i = 1</code>	i の値を 1 とする
2	<code>while i > 0:</code>	i > 0 である限り 3,4 行目を繰り返す
3	<code> print(i)</code>	i の値を表示する
4	<code> i+=1</code>	i の値を 1 増やす

このプログラムでは $i=1,2,\dots$ と順に i の値を一つずつ増やしながら処理を進めていきます。 $i > 0$ である限りは処理を繰り返すことからプログラムを実行したら（理論上は）ずっと動き続けることになります²⁶。また先に挙げた「コラッツの予想」のプログラムは、コラッツの予想が正しくなければ、反例となる n を指定したときに（理論上は）プログラムは停止しないことになります。なお OS の管理下で動作しているプログラムは OS によって強制的に停止させることはできます²⁷。

ここで見たような停まらないプログラムが書けてしまうことは問題に思えるかもしれませんが。しかし必ず停止することが保証できる「安全な」範囲にプログラムを

²⁶ 実際にはコンピュータで処理できるデータ量に限界があるため停止するはずですが、停止するまでには相当な時間がかかることでしょう(筆者も実際に試してみたことはありません)。

²⁷ この資料の最後に紹介する Spyder の場合、実行中のプログラムを強制終了するには[コンソール]メニューの[カーネルの再起動]を選びます。

制限してしまうと、そうでないときに比べて、プログラムで実現できる処理も制限を受けてしまうことが分かっています。

15.6 プログラミング演習の準備

ここではプログラミングを実際に始める方法の一つとして、本章でも取り上げているプログラミング言語 Python を対象として、プログラミングのためのパッケージのインストールの手順と、インストールされるツールを使ったプログラミングの方法について Windows を対象にしてごく簡単に説明します（MacOS や Linux でもおそらく同様に準備できるでしょう）²⁸。Python は比較的学習コストが低く、広く利用されていて、汎用性の高い言語です。またいわゆるデータサイエンスの分野で事実上の標準となっています。

さてプログラミングを始めるには「プログラミング環境」を準備する必要があります。プログラミング環境とは、プログラミングに関わるツール一式のことで、プログラムを編集するツール、プログラムを実行するためのツール、プログラム実行のインターフェイスとなるツールなどが含まれます。

Python については、プログラミング環境一式を提供するパッケージとして、さまざまな「ディストリビューション」がインターネットで配布されています。Python の代表的なディストリビューションの一つに Anaconda があります。Anaconda には数値計算、数式処理、データ解析、グラフ描画、画像処理、機械学習のための拡張機能やプログラム開発用のツールが最初から組み込まれています。またさらにパッケージを後から追加で導入するための仕組みも用意されています。

プログラミングを試してみるだけのためには Anaconda は大袈裟すぎるかもしれませんが。しかし Anaconda をインストールしておけば、大学での学習や研究でプログラミングを本格的に活用するための環境をごく簡単に準備できます。知っておいて損はないでしょう。

なおここでの説明は 2018-02-23 現在の情報をもとに作成しています。読者が参照している時点では、記載している情報が古くなっている場合もありえますので注意して下さい。

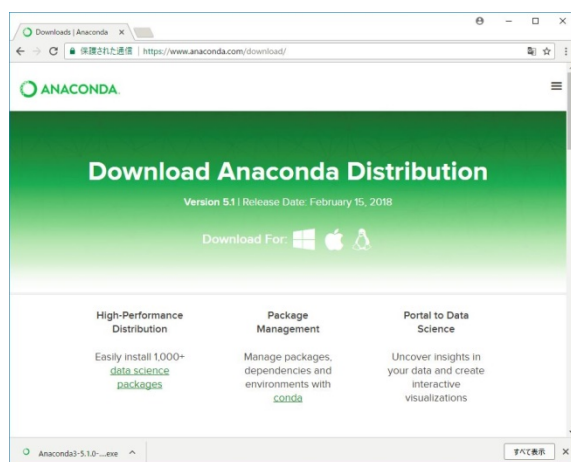
15.6.1 Python のインストール

まず Anaconda のパッケージをダウンロードして、インストールします。

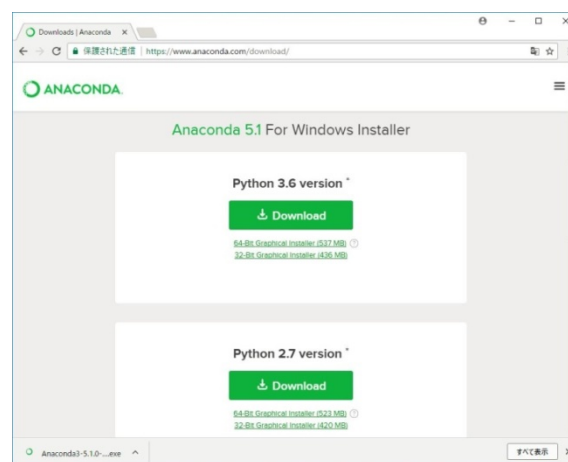
²⁸現行の最新版である Python3 を対象とします。一つ前のバージョンである Python2 も現状ではよく使われています。Python2 と 3 では文法に小さくない差異がありますので注意して下さい。

1) インストーラのダウンロード

まず WWW ブラウザで Anaconda の公式 WWW サイトのダウンロードのページ (<https://www.anaconda.com/download/>)を開きます (図 114(a))。ページを少し下にスクロールしたところにインストーラのダウンロードのボタンがあります。Python 3 のインストーラの「Download」ボタンをクリックします (図 114(b))。すると図 115 のような画面が表示されます。ここで Anaconda のドキュメントを入手する場合は Work Email と Role(職務)を入力して「Get the Starter Guide」を押します。インストール自体にはこのステップは不要です。「No Thanks」を押せば入力をスキップできます。



(a)



(b)

図 115 Anaconda 公式サイトのダウンロードページ

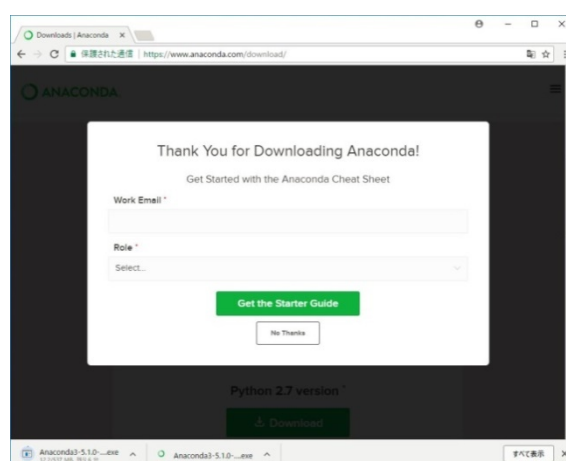


図 116 Anaconda インストーラのダウンロード画面

2) パッケージの確認(option)

ダウンロードしたインストーラを実行する前に、パッケージが不正なものではないことを確認すると安心です。具体的には例えばセキュリティソフトでインストーラをスキャンすることができます。また Anaconda のサイトに掲載されているハッシュ値(MD5,SHA256)を使ってパッケージの整合性を確認することもできます。

ハッシュ値とはファイルの指紋のようなもので、異なるファイルのハッシュ値は同一になることはありません。そこでダウンロードしたファイルのハッシュ値を計算したとき、その結果がサイトに掲載されているハッシュ値と一致すればファイルが正しいものであろうことが期待できます²⁹。

Anaconda のインストーラのハッシュ値の情報を調べるには、ダウンロードのページを下に辿ったところにある「How to Install ANACONDA」(図 116(a))の先のページ(図 116(b))から「Verify data integrity with MD5 or SHA-256」とリンクを辿って、

「Anaconda installer file hashes」のページを開きます(図 116(c))。「64-bit Windows, Py3」の先にハッシュ値を掲載しているページの一覧があります(図 116(d))。最新のインストーラの情報はリストの先頭にあります。インストーラのファイル名のリンクを辿ると、ハッシュ値の情報がありません(図 117)。SHA256 ハッシュを使うのがよいでしょう。

ダウンロードしたファイルの SHA256 ハッシュを知るには Windows PowerShell を使うとよいでしょう。Windows PowerShell は Windows に最初からインストールされているはずです。ここではインストーラとして「Anaconda3-5.1.0-Windows-x86_64.exe」を Downloads フォルダにダウンロードしたものとします。Windows PowerShell を起動したら、画面に次のように入力して最後に[Enter]キーを押します。

```
(Get-FileHash -Algorithm SHA256 Downloads/Anaconda3-5.1.0-Windows-x86_64.exe).Hash.ToLower()
```

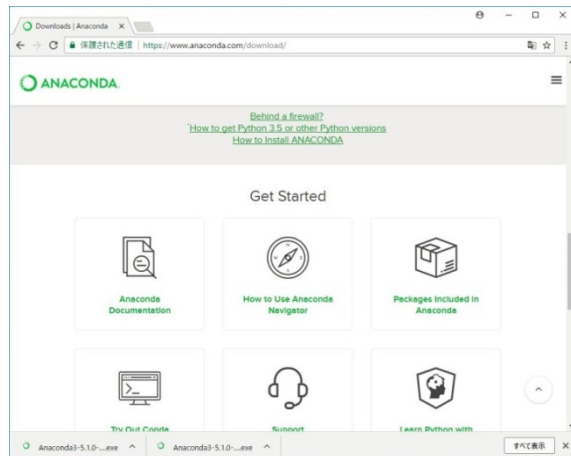
この結果として画面に表示される値(16進数)がサイトに掲載されている値と一致するかどうかを確認します。

3) インストールの実行

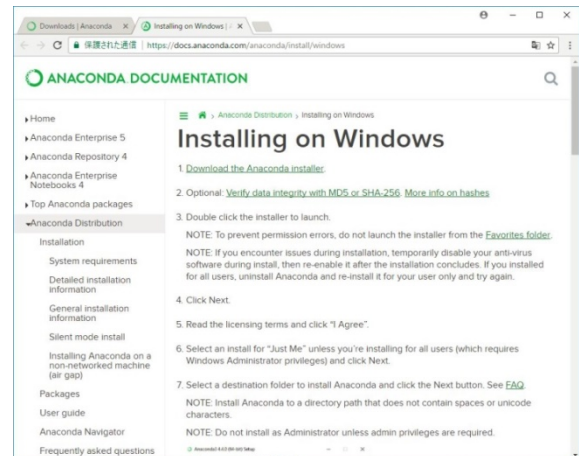
インストーラを起動したら、まず最初の画面(図 118(a))から「Next」で進んで、ラ

²⁹ 偽サイトにアクセスしていたり、サイトの情報が改ざんされていたりする危険性がゼロとは言えませんが、ハッシュ値が一致したからといって 100%安心とまでは言い切れません。

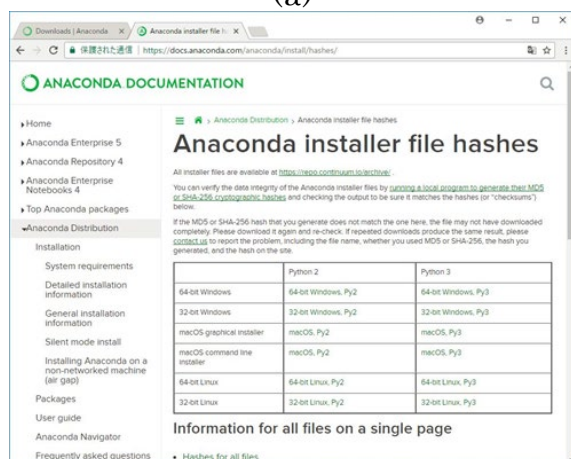
イセンスを確認します(図 118(b))。ライセンスに同意してインストールを続行するなら「I Agree」を選んで進みます。次にインストール方式を設定します(図 118(c))。ここは「Just Me」のままでよいでしょう。つづいてインストール先のフォルダを設定します(図 118(d))。これも基本的にはそのままで変更しなくてよいでしょう。なおここでインストールのために必要なストレージの容量、利用可能な空き容量も確認できます。フォルダを指定できたら「Next」で進みます。



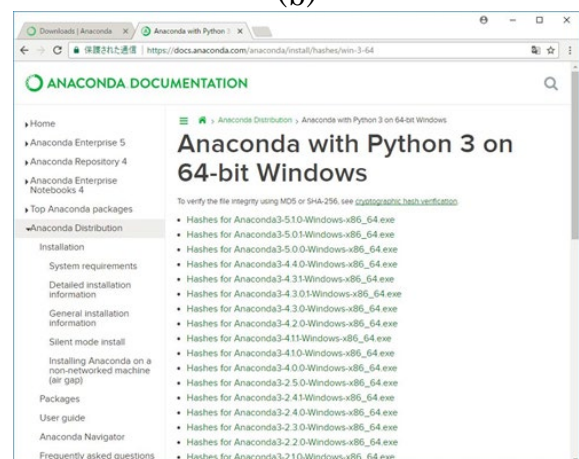
(a)



(b)



(c)



(d)

図 117 インストーラのハッシュ値のページへ

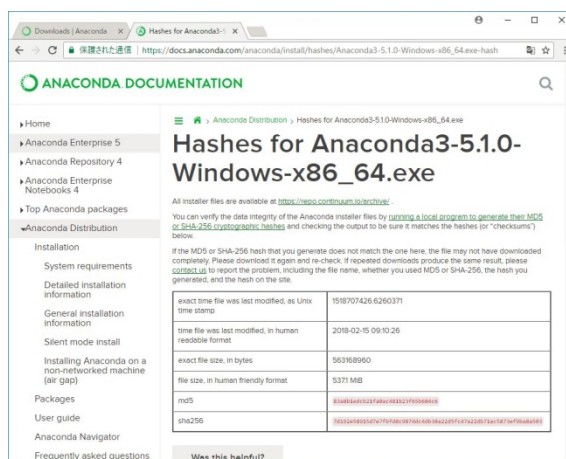
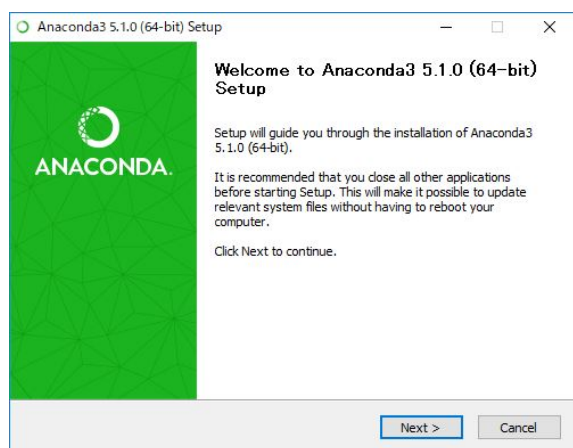
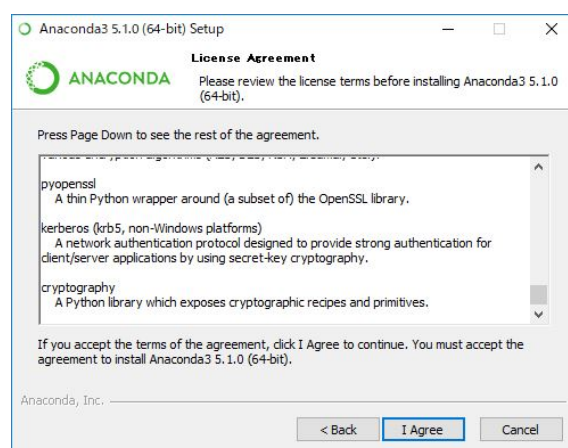


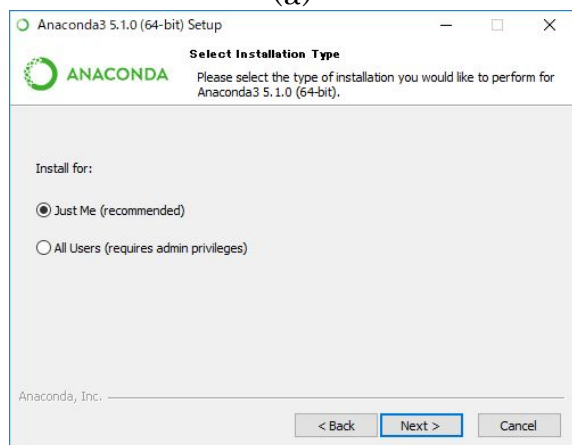
図 118 インストーラのハッシュ値



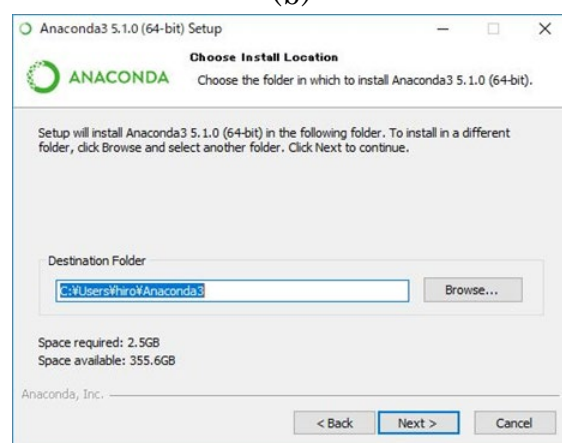
(a)



(b)



(c)

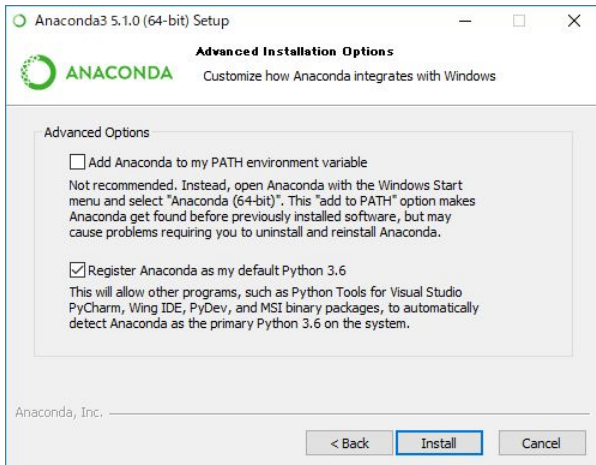


(d)

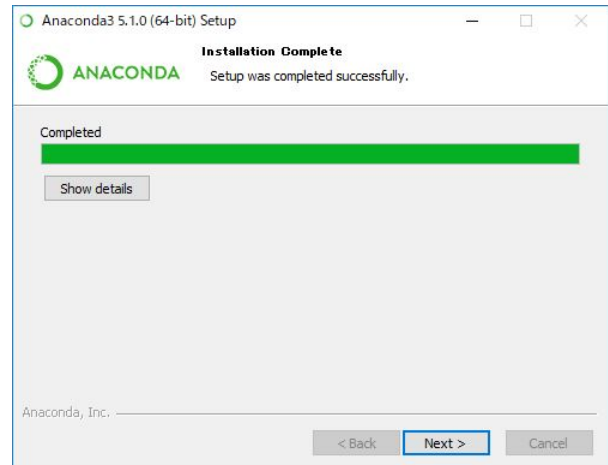
図 119 Anaconda のインストール(前半)

次の「Advanced Options」(図 119(a))の設定は「Register Anaconda as my default Python 3.6」のままでよいでしょう。ここで「Install」を押せばいよいよインストールが始ま

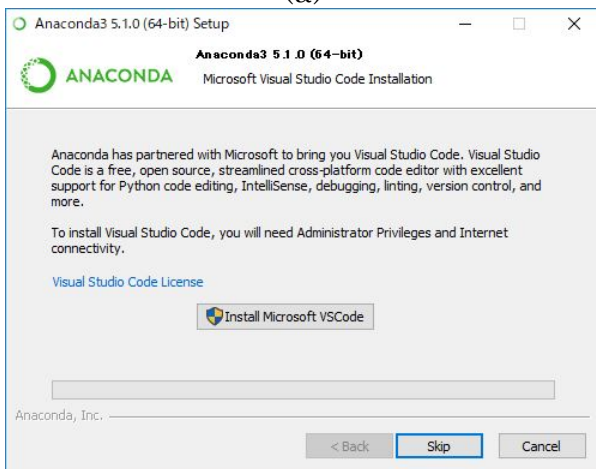
ります。インストールにはしばらく時間がかかります。何も問題がなければ完了画面(図 119(b))に移ります。ここから「Next」で「Microsoft Visual Studio Code Installation」というツールの追加インストールに移ります(図 119(c))。これは Anaconda に直接には関係ないためスキップしても構いません。次がようやく最後の画面です(図 119(d))。ここで関連情報を確認することもできますが、画面上のチェック ☒ を二つとも外して「Finish」を押せば、そのままインストールが終了します。インストール後には Windows のアプリの一覧に「Anaconda」が追加されているはずです。



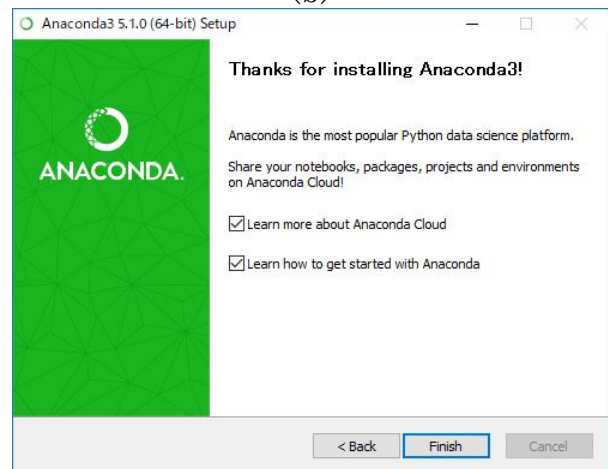
(a)



(b)



(c)



(d)

図 120 Anaconda のインストール(後半)

15.6.2 Python を使ってみる

インストールが完了したら Python のプログラムを作成、実行してみましょう。Anaconda のメニューの中に「Spyder」という項目が見つかるはずです。起動すると図 121 のような画面が出てきます。Spyder はプログラムを編集、実行、デバッグするための機能をまとめて提供するツールです。一般にこのようなプログラミングのための「全部入り」のツールを IDE(Integrated Development Environment; 統合開発環境)

と呼びます。さまざまなプログラミング言語にさまざまな IDE が用意されています。

Spyder のウインドウの左半分はエディタ画面でプログラムのファイルを編集するために使います。最初は「temp.py」という仮の名前のついたファイルが開かれた状態になっています。

ウインドウの右下はプログラムを実行するコンソール(console;操作卓)画面です。最初は「コンソール 1/A」というタブが開かれている状態になっています。コンソー

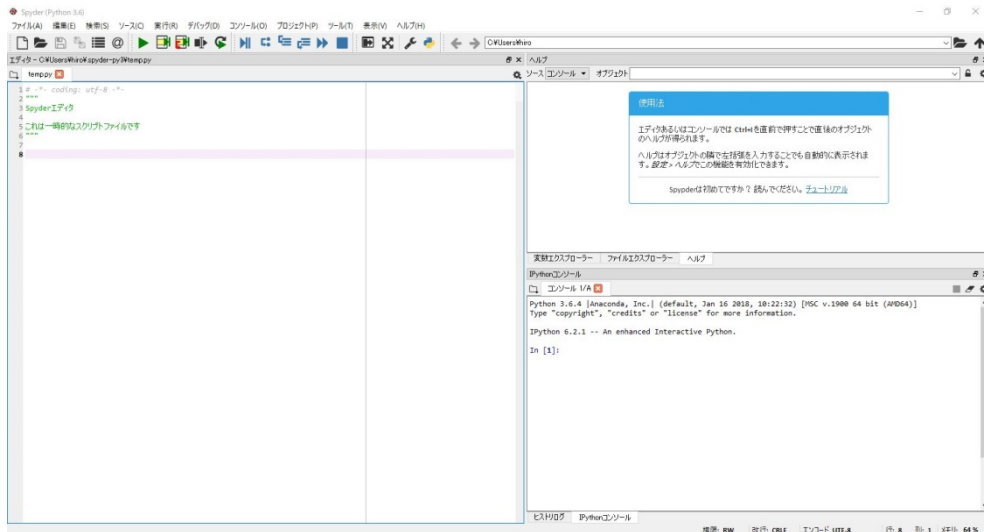


図 122 Spyder の初期画面

ル画面では IPython(Interactive Python)というツールが動作しています。

IPython では Python のコード(式や文)を一つずつ処理するようになっています。最初に画面に表示されている「In [1]:」はプロンプトとって、コンソールがコードの入力を受け付けていることを示しています。ここでプロンプトにつづいてコードを入力して[Enter]キーを押すと、コードがその場で処理されて結果が表示されます。処理の結果は「In [1]:」に対応して「Out [1]:」とともに表示されます。ここで画面には次のプロンプト「In [2]:」が表示されているはずです(図 120)。このようにプロンプ

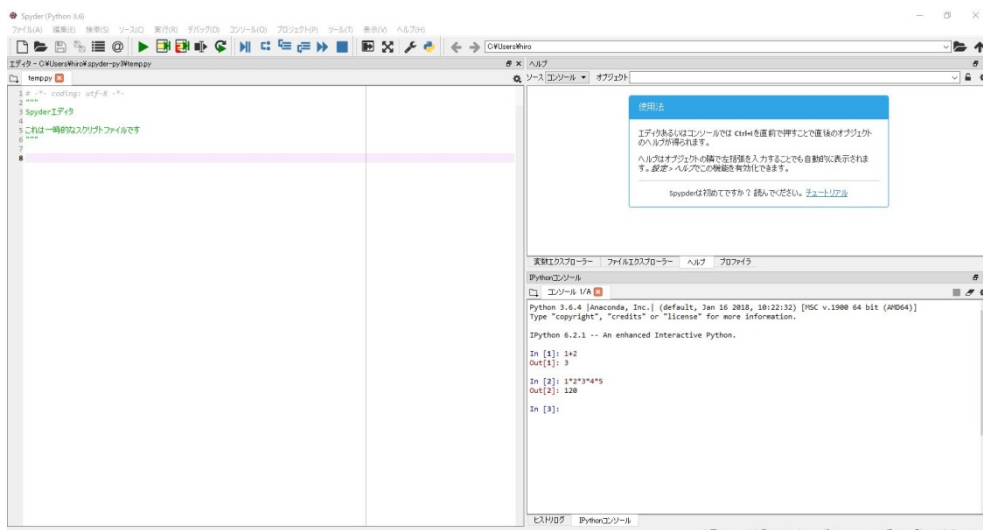


図 121 コンソールでのコードの評価の例(Spyder)

トには、処理したコードの数の情報が含まれます。

Spyder のウインドウの右上の画面には「ヘルプ」「ファイルエクスプローラー」「変数エクスプローラー」というタブが配置されています。「ヘルプ」はプログラムに関わるさまざまな情報検索に使います。「ファイルエクスプローラー」は Windows のエクスプローラーと同様のものです。「変数エクスプローラー」はコンソールに入力されたコードに関する情報を表示する機能を提供します。

それでは Spyder を使って本資料に掲載している「コラッツの予想」のプログラムを作成、実行してみましょう。まず「ファイル」メニューから「新規ファイル」を選択して、適当なフォルダにファイルを作成して下さい。ファイル名は「collatz.py」としましょう³⁰。ここでエディタの画面に「collatz.py」という新しいタブが開くはずですのでリスト 5 の通りにプログラムを入力して下さい。

Spyder のエディタで Python のプログラムを作成していると、改行したときに次の行が自動的に字下げされ、行頭に空白が入る場合があることに気づくでしょう。そのような処理が行われるのは Python では字下げ(インデント)によってプログラムの構造が定まるためです。行のインデントが正しくないとプログラムは動かなくなります。なおインデントを浅くすべきところでは必ずしも自動的に処理はされません。インデントを手動で浅くするには[BackSpace]キーを押して下さい。

リスト 5 コラッツの予想を試すプログラム（再掲）

行番号	プログラム	解説
1	<code>n = int(input("n? "))</code>	キーボードから n の値を得る
2	<code>while n != 1:</code>	n ≠ 1 である限り 2～7 行目の処理を繰り返す
3	<code> print(n,end = " -> ")</code>	途中経過の表示(現在の n の値)
4	<code> if n % 2 == 0:</code>	n が偶数(2 で割って余りが 0)のときは
5	<code> n = n // 2</code>	→n を 2 で割った値を再び n とする
6	<code> else:</code>	そうでないとき(n が奇数のとき)は
7	<code> n = 3*n+1</code>	→n を 3 倍して 1 を加えた値を再び n とする
8	<code>print("1 OK")</code>	ここに至った時点で n=1 のはず

プログラムがすべて入力できたら、実行してみましょう。プログラムを実行するには、「実行」メニューから「実行」を選びます。ここで「実行設定」画面が表示さ

³⁰ 「.py」は Python ファイルを表す拡張子として一般的に使われます。

れた場合には、そのまま「実行」を選んで構いません(図 123)。「コラッツの予想」のプログラムの実行が開始されると、コンソール画面に「n?」と表示されます(図 122)。ここで2以上の整数を指定して[Enter]キーを押すと計算が始まって、処理の過程が順に表示されていって、(コラッツの予想が正しければ)ほどなく処理が終了します。



図 124 実行方法の設定(Spyder)

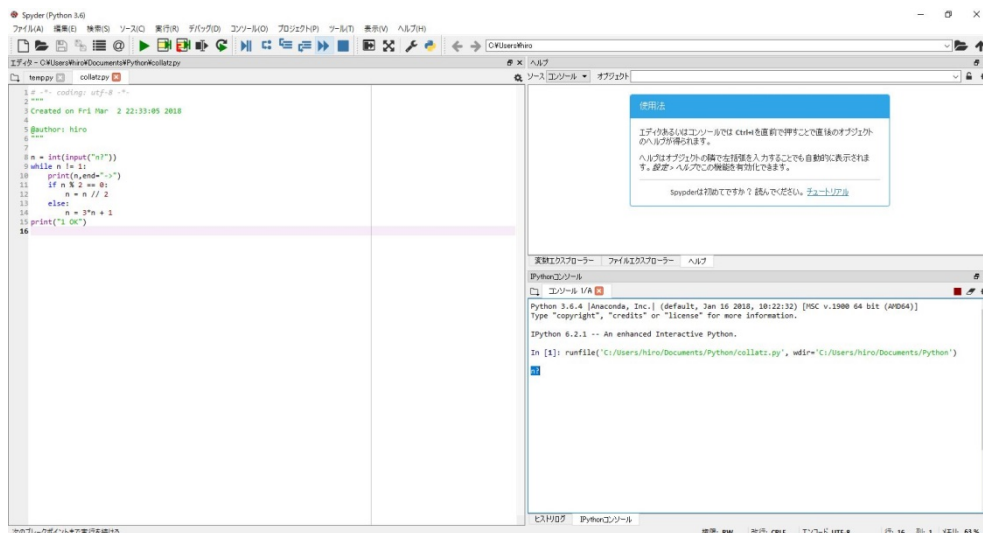


図 123 コラッツの予想の実行(n の入力)

最後にプログラムの挙動を変えてテストしてみることにします。次のリスト 6 のようにプログラムの 3 行目と 8 行目を書き換えて保存した上で、

さきほどと同様に実行してみてください（変更部分は赤字で示しています）。

リスト 6 コラッツの予想のプログラム（データ出力形式を変更したもの）

行番号	プログラム	解説
1	<code>n = int(input("n? "))</code>	キーボードから n の値を得る
2	<code>while n != 1:</code>	n ≠ 1 である限り 2～7 行目の処理を繰り返す
3	<code> print(n)</code>	途中経過の表示(現在の n の値)
4	<code> if n % 2 == 0:</code>	n が偶数(2 で割って余りが 0)のときは
5	<code> n = n // 2</code>	→n を 2 で割った値を再び n とする
6	<code> else:</code>	そうでないとき(n が奇数のとき)は
7	<code> n = 3*n+1</code>	→n を 3 倍して 1 を加えた値を再び n とする
8	<code>print(1)</code>	ここに至った時点で n=1 のはず

今度は数値が 1 行に 1 つずつ表示されるはずです。たとえば処理が終了するまでの過程での値の変化を Excel や Calc 等のツールを使ってグラフにして解析しようとするのであれば、このようにデータを表示した方が便利でしょう。またこのようにプログラムを書き換えてみることで、コンピュータがプログラムに従ってデータ処理を行っていることが実感できるでしょう。

プログラミングのスキルを身につけるには、とにかく試してみることが大切です。本資料をきっかけとして、プログラミングの世界の住民になってもらえればうれしく思います。

参考文献

- [21] 鎌田 正浩: 確かな力が身につく Python「超」入門, SB クリエイティブ (2016/3/16) ISBN: 978-4-797384406
- [22] Guido van Rossum (著) 鴨澤 真夫 (訳): Python チュートリアル 第3版, オライリージャパン (2016/3/24) ISBN: 978-4-873117539

16. あとがき

この教科書ではアカデミックな活動での ICT 利用スキルについて学びました。1.1 節でも述べたように、この教科書で学んだことから、さらにご自身に興味や専門的な分野の必要性に応じて様々な ICT 利用スキルを主体的に獲得していくことが求められます。本書の末尾の別表 1 に示した評価基準もこのことを意図して作成しています。さらにそれら以外にも（すべてを身に着ける必要もないかと思いますが）、以下のようなスキルの獲得も考えられます。

- オンラインストレージなどの効果的な活用
- パーソナルコンピュータとスマートフォンと連携利用
- 英語、日本語以外の言語での文字入力
- 文献管理ソフトの利用
- データベースの構成と利活用
- ビデオの編集と利活用
- CAD や 3 次元プリンタの利用
- Web ページの作成とプログラミング
- SNS などのソーシャルメディアの効果的な利用
- プロジェクトの管理とそのためのツールの利用

これらのスキルを学んで行くための学習資源を探索したり、共に学んだり、教えあったりする仲間のネットワーク作りも重要な視点です。

コンピュータを中心とする情報通信技術の特徴は自身で新しいツールを創出できることです。本書ではコラムとして **Great Idea!** を設けて情報通信技術は誰かが素晴らしいアイデアを得て作り上げてきたものであることを紹介しました。将来、皆さんの貢献が **Great Idea!** として紹介できるなら、それは素敵なことだと思いませんか。

付録 略語集

コンピュータやネットワークを利用しているとさまざまな略語が使われます。ここでは主なものを紹介します。

略語	綴り	意味
情報関連の単位		
bit		2 進数1桁の情報量を表す
byte		2 進数8桁の情報量を表す
bps	bit per second	情報を伝送する速度, 1秒間に送るビット数
byte/sec		情報を伝送する速度, 1秒間に送るバイト数
k	kiro	10 の3乗 (IS単位系の補助単位)
M	mega	10 の6乗 (IS単位系の補助単位)
G	giga	10 の9乗 (IS単位系の補助単位)
T	tera	10 の12乗 (IS単位系の補助単位)
P	peta	10 の15乗 (IS単位系の補助単位)
文字コードとキーボード		
ASCII	American Standard Cord for Information Interchange	アルファベットや数字などを表す米国で制定された文字コード
JIS X0201	Japanese Indsutrial Standard	ASCII コードに準じたアルファベットや数字にカタカナを加えた JIS 規格の文字コード, いわゆる半角カナコード, 記号の一部が ASCII コードと異なっている
JIS X0208		2バイトをもちいて漢字を収容したコード
Shift_JIS		JISX0201(いわゆる半角文字)と JISX0208(いわゆる全角文字)を一括して扱うための文字コードの表現法
Unicode		世界で用いられている様々な文字コードを共通に利用できるようにしたコード
UTF-8		unicode の表現方法の一つで1文字に1~4バイト使用する. ASCII に対して上位互換となっている
ASCII 配列		キーボードでの記号類の配列として米国で用いられている配列. US 配列
JIS 配列		キーボードの記号類の配列として日本で標準的に用いられている配列
Ctrl, Alt, Esc, Ins,	Control, Alternate, Escape, Insert の略	

ISO2022-JP		インターネット上で日本語による通信のために策定された文字コードで、JIS コードともよばれる
EUC-JP	Extended UNIX Code Packed Format for Japanese	UNIX で扱われる文字コード(EUC)を日本語に対応させたもの
QWERTY 配列		キーボードの下から3段目の文字が左から QWERTY の順に並ぶ文字配列のこと

ファイルのフォーマットと拡張子

HTML	Hyper Text Markup Language	Web ページなどを書式を含めて記述する言語
XML	eXtensible Markup Language	汎用に記述できるマークアップ言語
PDF	Portable Document Format	Adobe 社が開発した可搬性のある文書のフォーマット
TXT		text を略した拡張子
JPEG	Joint Photographic Experts Group	静止画像のフォーマットの一つ。画質を落としてデータ量を削減する非可逆圧縮のフォーマットとして使われている。
BMP	Microsoft Windows Bitmap Image	画像ファイルのフォーマットの一つ。圧縮せずにもちいられる場合が多い。
GIF	Graphic Interchange Format	画像ファイルフォーマットの一つ。256 色以下の画像を可逆圧縮する
PNG	Portable Network Graphics	画像ファイルフォーマットの一つ。画像を可逆圧縮する
ZIP		複数のファイルを1つにまとめるフォーマットのひとつ
MP3	MPEG1/2 Audio Layer-3 の略	音声ファイルのフォーマットのひとつ。
MP4		動画ファイルのフォーマットのひとつ
MPEG	Moving Picture Expert Group の略	動画のフォーマットを検討しているグループ、ここで提唱されたフォーマットとして MPEG-1, MPEG-2, MPEG-4 などがある
EXE	Executable の略	windows のアプリケーションソフトなど実行ファイルのフォーマット
ISO	ISO は国際標準化機構 (International Organization for Standardization)	ISO9990 形式の光学ディスク(CD, DVD など)の内容を一括したフォーマット
EPUB	electronic publication	電子書籍フォーマットのひとつ
DOC, DOCX, XLS, XLSX, PPT, PPTX		ワード、エクセル、パワーポイントのファイルの拡張子、X がついているものは xml 形式で表現されている
wav, aiff		音声ファイルのフォーマット

AVI	Audio Video Interleave	動画ファイルのフォーマットの一つ。Windows での標準的なフォーマット
MOV		動画ファイルのフォーマットの一つ。主に Apple QuickTime Player で利用される
FLV	Flash Video	動画ファイルのフォーマットの一つ。Adobe Flash Player でサポートされる
CSV	Comma Separated Values	データベースや表計算ソフト内のデータについて各行のセルに含まれる数値や文字列がカンマ (,) で区切られたファイルの形式。異なるアプリケーション間でのデータのやり取りが容易になる
Exif	Exchangeable image file format	画像ファイルのフォーマットの一つ。デジタルカメラの機種や撮影日時等のメタデータが保存される
SVG	Scalable Vector Graphics	画像ファイルのフォーマットの一つ。XML で記述されており、ウェブサイト上で画像に動きをつけること等ができる
TIFF	Tagged Image File Format	画像ファイルのフォーマットの一つ
MIDI	Musical Instrument Digital Interface	異なる機器間で音楽の演奏データ等をやり取りするための国際規格
LaTeX		数式の組版等に優れた論文等の学術文書を作成するためのツール

記録メディア

SD card, Mini SD, Micro SD		
CD	Compact Disc	
DVD	Digital Versatile Disc	
BD	Blu-ray Disc	

PC の構成要素

PC	Personal Computer	パーソナルコンピュータ
CPU	Central Processing Unit	中央処理装置
GPU	Graphic Processing Unit	グラフィクスを処理する装置
HDD	Hard Disk Drive	磁気ディスクのドライブ
SSD	Solid State Drive	フラッシュメモリを用いたドライブ
BIOS	Basic Input Output System	PC の起動などのために読み出し専用メモリに書きこまれたプログラム
OS	Operating System	オペレーティングシステム
NTFS	NT File System	Windows が採用するファイルシステムの一つ
FAT	File Allocation Table	ファイルシステムの一つ。macOS でも読み書きができる

PC のインターフェイス

USB	Universal Serial Bus	
VGA	Video Graphics Array	VGA規格の端子, RGB のアナログ信号を用いている。
HDMI	High Definition Multimedia Interface	映像と音声を一括して送受するケーブルや端子の規格。デジタル信号を用いている。
DVI	Digital Video Interface	
Thunderbolt		
Bluetooth		
DisplayPort		
PCIe	PCI Express, PCI は Peripheral Component Interconnect	
IEEE 1394/Firewire/i.LINK		
NFC	Near Field Communication	IC カードやスマートフォンなどで使われる数センチメートル程度の近距離で用いる無線通信の規格。
VGA, SVGA, XGA, WXGA, HD FWXGA, SXGA		
RGB	Red Green Blue	光の三原色
CMYK	Cyan Magenta Yellow Key plate	プリンタインクなど減法混合のための原色(シアン, マゼンタ, 黄色)に黒を加えたもの

ネットワーク関連

LAN	Local Area Network	建物内, 事業所内などの狭い範囲に敷設するネットワーク
WAN	Wide Area Network	広域のネットワーク
FTTH	Fiber To The Home	一般個人向けに光ファイバ回線で高速のデジタル通信を行うこと
CATV	cable television	ケーブルテレビ, テレビ番組の配信とともにインターネット接続のサービスにも用いられる
ADSL, VDSL	Asymmetric Digital Subscriber Line Very high bit rate DSL	電話回線 (subscriber line) を用いたデジタル通信の方式
PHS, 3G, LTE		携帯電話回線の方式
WiMAX		
ONU	Optical Network Unit	FTTH で光ファイバの接続を受ける通信ユニット
WiFi	wireless fidelity	無線 LAN のこと

IEEE 802.11(a,b,g,n,ac),		無線 LAN の規格, 最後のアルファベットは通信速度や使用する周波数帯域ごとの規格を示す. IEEE は The Institute of Electrical and Electronics Engineers の略, 米国の電気電子工学の学会
WEP	Wired Equivalent Privacy	無線 LAN の暗号化方式の規格
WPA, WPA2	Wi-Fi Protected Access	無線 LAN の暗号化方式の規格
WPS	Wifi Protected Setup	無線 LAN 機器同士の暗号化設定を容易にする規格
AOSS	AirStation One-touch Secure System	無線 LAN 機器同士の暗号化設定を容易にするバッファローの方式
Ethernet, 10Base-T, 100Base-TX, 1000Base-T		有線 LAN の規格, 数字は通信速度(bps)
UTP	Unshielded Twisted Pair	イーサネットで用いられるツイストペア線のうち, 全体を導体でシールドしていないもの
IP	Internet Protocol	インターネット上でパケット(データグラム)を配送するためのプロトコル, パケットの到着は保証しない. TCP とともに利用されることが多いことから TCP/IP という表現が行われる
TCP	Transmission Control Protocol	データの順序を維持して連続した通信を IP 上で実現するためのプロトコル
UDP	User Datagram Protocol	IP で行われるパケット(データグラム)単位の通信をユーザプログラムから利用するためのプロトコル
DHCP	Dynamic Host Configuration Protocol	コンピュータの IP アドレスをサーバから提供を受けて動的に付与するためのプロトコル
NTP	Network Time Protocol	ネットワークを介してサーバから時刻の提供を受けるプロトコル
DNS	Domain Name System	IP アドレスの代わりにドメイン名でアクセスするための仕組み
HTTP	Hyper Text Transfer Protocol	WWW のページへのアクセスのためのプロトコル, TLS/SSL で保護したものは HTTPS と書く
PPTP	Point to Point Tunneling Protocol	VPN を構成するためのプロトコルの一つ
SMTP	Simple Mail Transfer Protocol	メールサーバ間のメッセージの配送, メールソフトからメールサーバへのメッセージの送出に用いられるプロトコル. TLS/SSL で保護する場合は SMTPSs と書かれる
POP	Post Office Protocol	メールサーバからメッセージの提供を受けるプロトコル, 現在バージョン3が用いられており POP3 と書かれる, また TLS/SSL で保護する場合は POP3s と書かれる

IMAP	Internet Message Access Protocol	メールサーバからメッセージの提供を受けるプロトコル, 現在バージョン4が用いられており IMAP4 と書かれる, また TLS/SSL で保護する場合は IMAP4s と書かれる
TLS, SSL	Transport Layer Security Secure Sockets Layey	WWW の通信などを暗号化して保護するための通信方式
FQDN	Fully Qualified Domain Name	DNS での省略のない形で記載されたドメイン名
WWW	World Wide Web	インターネット上でハイパーリンクにより相互に参照されるページを提供する仕掛け
SNS	Social Netowrk Service	特定の知人間のコミュニケーションを支援するサービス
URL	Uniform Resource Locator	WWW など Web サイトの所在などを一定のルールで記載したもの
ISP	Internet Service Provider	インターネット接続サービスの提供者
VPN	Virtual Private Network	インターネット上で暗号化通信などで保護して特定の組織用のネットワークを構成する技術
NAS	Network Attached Strage	ネットワークに接続す形式のディスクドライブなどのストレージ
SMS	Short Message Service	電話番号あてに携帯電話間で送受する短いテキストメッセージの配信サービス
To, Cc, Bcc,	cc は carbon copy, bcc は blind carbon copy	電子メールの宛先を書くためのヘッダ
DoS, DDoS	Denial of Service Distributed DoS	サーバなどの機能を妨害する攻撃, 複数のコンピュータから行う攻撃を DDoS 攻撃という
IoT	Internet of Things	モノのインターネット, さまざまなモノがネットワークに接続されること
M2M	Machine to Machine	機械と機械が直接ネットワークで接続される. IOT と同様の概念
CPS	Cyber Physical System	仮想空間と実世界の装置などが結合したシステム
P2P	Peer to Peer	末端のコンピュータどうしがサーバなどを介さず直接やり取りするシステム
SSID, ESSID	(Extended) Service Set IDentifier	個々の無線 LAN グループを識別するために付けられている名前
TKIP	Temporal Key Integrity Protocol	無線 LAN の暗号化に関する規格 WPA で採用されている暗号化方式
IPv6	Internet Protocol Version 6	IP のバージョン 6. IPv6では IP アドレスを16バイト(128ビット)で表現できるため, 割り当て可能なアドレスが枯渇している IPv4 の後継としての移行が進みつつある
ARP	Address Resolution Protocol	ネットワーク上で IP 通信を行う際, IP アドレスから受け手側の MAC アドレスを特定するためのプロトコル

NAT	Network Address Translation	インターネット通信時にプライベート IP アドレスをグローバル IP アドレスに変換する技術のこと
MAC アドレス	Media Access Control Address	LAN の機器ごとに予めメーカーによって割り振られた固有の番号. 48 ビットを 1 バイトごとに 16 進数 2 桁で表した数値を「-」や「:」でつなぎ、「〇〇-〇〇-〇〇-〇〇-〇〇-〇〇」と表記される
VDI	Virtual Desktop Infrastructure, 仮想デスクトップ基盤	それぞれの利用者の端末で行う処理をサーバ内の仮想化された PC に一元化し, 利用者は端末に表示された画面上で入出力のみを行う環境を実現するための基盤
IKEv2	Internet Key Exchange Version 2	VPN 接続技術の一つで, 本人認証や暗号化等により通信する相手側と安全な接続を確保するためのプロトコル
AES	Advanced Encryption Standard	共通鍵暗号方式の一つ

その他		
GPS	Global Positioning System	人工衛星を利用した測位システム
Felica		ソニーが開発した IC カード, 交通カードや本学の学生証に利用されています
QR Code		2次元バーコードの1種
IR	Infra-Red	赤外線, 紫外線は Ultra Violet (UV)
CGM	Consumer Generated Media	消費者生成メディア
FAQ	Frequently Asked Question	よくある質問
CC	Creative Commons	マルチメディアコンテンツの流通を促すために, 作者があらかじめクレジット表示や改変等についての利用条件をコンテンツ内に示しておく仕組みのこと
ICT	Information and Communication Technology	情報通信技術
CAD	Computer-Aided Design	コンピュータを用いて機械などを設計すること。またそのためのソフトウェア

大学・学術利用の情報システム等		
IR	Institutional Research	大学などが自分自身の組織について調査すること
HPC	High Performance Computing	スーパーコンピュータなどを利用する高性能計算
LMS	Learning Management System	学習管理システム, Course Management System と類似したシステム
CMS	Course Management System	コース管理システム (例えば京大の PandA)
CMS	Content Management System	Web のページを管理するためのシステム

MOOC	Massive Open Online Courses	京大が提供する EdX など公開のオンライン講座
SPOC	Small Private Online Courses	MOOC と同様の技術を学内用などに使うオンライン講座

京都大学や学術機関の情報サービス

KUINS	Kyoto University Integrated Network System	京都大学の有線、無線LANなどのネットワークです
KULASIS	Kyoto University's Liberal Arts Syllabus Information System	履修登録やシラバス閲覧をするための教務情報システムです
KUMOI	Kyoto University Mail cIoud Interface	学生メールシステムの愛称です
KULINE	Kyoto University Libraries Information NEtwork system	附属図書館の蔵書検索システムです
KURENAI	Kyoto University Research Information Repository	論文などを公開する京都大学のリポジトリです
PandA	People and Academe	京都大学のコース管理システムです
Gorilla	Global Online Resources for International Language Learning Assistance	国際高等教育院の語学学習用のシステムです
CiNii	Citation Information by NII	国立情報学研究所(National Institute of Informatics) の学術情報データベースです
eduroam		欧州で開発された国際的な学術無線 LAN の仕組み. eduroam に加盟している他の高等教育機関等での無線 LAN 接続が可能である
ECS-ID	Educational Computer System ID	京都大学が学生向けに提供しているアカウント
SPS-ID	School Personnel System ID	京都大学が教職員向けに提供しているアカウント

別表 1 評価基準

別表 情報基礎演習 評価基準

本評価基準は情報基礎演習で扱う項目について学習者の達成度を評価するための基準である。レベルはこの科目の合格レベル（C）， 達成目標レベル（A）， 在学中に身につけて欲しいレベル（S）の3段階とした。背景が白色のものは扱わない

大項目	中項目	小項目	レベルC	レベルA	レベルS	備考
			必要な技能を不十分ながら獲得できている(演習科目合格レベル)	必要な技能を基本的なレベルで獲得できている(演習科目達成目標レベル)	在学中に身につけて欲しいレベル	実施の考慮事項と評価方法
学内情報サービスとネットワークの利用						
	学内情報サービス		学内で提供されている情報サービスを理解しているが一部のみ利用できる	学内で提供されている情報サービスを理解し利用できる	学内で提供されている情報サービスを効果的に利用して学習に役立てている	履修登録、学習管理システム(LMS)、学内ネットワークなどの利用で評価
	私物のP Cなどの学内ネットワークへの接続		VPN等大学が提供するサービスを利用してP Cやスマートフォンなどを援助者の支援を得て大学のネットワークに接続できる	接続における暗号化などを理解し、VPN等大学が提供するサービスを利用してP Cやスマートフォンなどを大学のネットワークに接続できる	機器の特性や利用環境に応じて複数の接続方式を利用できる	実技での評価
	電子メールを用いたフォーマルなコミュニケーション					
		適切な内容のメールの作成	件名、差出人、受信者を明示したメールを書ける	件名、差出人、受信者を明示し、適切な内容と言葉遣いのメールを書ける	Cc, Bcc, Reply-to などのフィールドを適切に利用してグループでのコミュニケーションに活用できている	メールの利用演習
		受け取ったメッセージの正しい取扱い	受け取ったメールのメッセージの適切な扱いを理解し、実践できている	同左	同左	テストなどで評価
		添付ファイル利用時の配慮	サイズが適切な添付ファイルの送受はできるがサイズの圧縮や暗号化はできない	適切なサイズ、形式の添付ファイルを送受できる。重要な内容の添付ファイルの暗号化ができる。	ファイルを送受する他の手法とその安全な利用を理解し、メールと組み合わせる適切に利用できる	メール利用演習
	授業で利用する コース管理システム などについての操作		LMSを用いたファイルの閲覧や課題の提出は行えるが教員の指示には十分に從っていない	LMSを用いたファイルの送受、課題の提出などを適切に行える	履修登録システム, LMS, 学生用メールなどを適切に連携させて利用できる	LMS(PandA)の利用で評価
	情報ネットワークを利用する上での注意		情報ネットワークを利用する上での注意事項はある程度は知っているが、実践は不十分である	情報ネットワークを利用する上での注意事項を知り、適切に実践している	情報ネットワークを利用する上での注意事項について主体的、継続的に情報を獲得し、実践を改善している	情報セキュリティ e-Learning の修了
大学における知的生産（梅棹忠夫の造語）と ICT 活用			大学での授業や課外活動などでの ICT 活用シーンを理解しているが活用は限定的である	大学での授業や課外活動などでの ICT 活用シーンを理解し、活用している	大学での授業や課外活動などで ICT を積極的に活用している	以下の各項目で評価
パーソナルコンピュータとOS、その構成と基本操作						
	自立したパソコン所有者として、コンピュータやOSについての基本的な理解と安全な運用（セキュリティ、ソフトウェアのライセンス）について学ぶ。					
		コンピュータの仕組みとO Sの役割の理解	コンピュータの仕組みやO Sの役割の基本的な事項をある程度知っているが、安全な設定で利用するには他者の支援を要する	コンピュータの仕組みやO Sの役割の基本的な事項をしており安全な設定で利用できている	コンピュータの仕組みやO Sの役割の基本的な事項を知っており、トラブル対策や性能向上などをはかることができる	テストなどで評価
		ソフトウェアのライセンスと適切な利用	ソフトウェアのライセンスをある程度理解し、問題ない程度に利用している	ソフトウェアのライセンスを理解し、適切な状況で使用している	ソフトウェアのライセンスを理解し、適切な状況で大学の所有するライセンス、オープンソースのソフトウェアなどを使用できる	同上
		P Cへの外部機器等の接続	P Cのネットワークへの接続や、プリンタや外部記憶装置を接続には他者の支援を要するが、運用はできる	自分自身でP Cをネットワークに接続したり、プリンタや外部記憶装置を接続して利用できる	有線・無線L A Nなどを適切に設定してP Cやプリンタなどを接続することができる	同上
	アプリケーションを利用する基盤としてのOSにまつわる操作（ファイル、デバイス（プリンタやネットワーク）、プロセスとメモリ、ユーザ、起動と停止、GUI）を理解する。OS としてはWindows と Linux のいずれかオプションとなる。		O Sが管理している事項について必要な設定等を補助者の支援を得て行える	O Sが管理している事項について必要な設定等が自分自身で行える	P Cを効果的・効率的に利用するためにO S等の設定を行える	同上
	タッチタイピング、ショートカットキー、英文タイピングの基礎		1）タッチタイピングの指使いを理解し、ある程度実施している。 2）ショートカットキーを利用している。 3）英文のタイピングルールを知り、実践している。 4）適切な文字コードで入力できる。	1）タッチタイピングの指使いを理解し、実施している。 2）～4）同左	1）手元を見ることなくタイピングができる。 2）マウスが使えない状況でもある程度のP Cの操作が可能である。 3）同左 4）適切な文字コードを使用するとともに必要に応じて変換できる。	タイピング速度、英文で作成した文書
情報探索と図書館利用			大学図書館等が提供している情報源について理解し、必要な情報がある程度探索できる	大学図書館等が提供している情報源について理解し、必要な情報を網羅的・系統的に探索できる	大学図書館等が提供している情報源について理解し、必要な情報を効果的、効率的に探索し、批判的に利用することができる	テストや、検索演習で評価
表計算ソフトウェアによる情報の加工						
	機械処理に適したデータの表現、入力値の制限方法、データの検査	機械処理に適した表を構成できる。	機械処理に適した表を構成できる。	レベルAに加え、入力するデータの検査を効果的に実施できる	表計算ファイル	
	データ操作：データの並べ替え（ソート）や単純／クロス集計といったデータ分析の基本	データを並べ替えて分析することができる。単純集計が行える	レベルCに加え、クロス集計が行える。	レベルAに加え、データ分析の必要に応じて、複合的な条件付きの集計などが行える	表計算ファイル	
	表計算ソフトを用いた数値シミュレーション	与えられた差分方程式のシミュレーションが行える	パラメータを変化させたシミュレーションを行える	自らの必要に応じてシミュレーションを構成できる	同上	
	データのグラフへの可視化、学術的なレポートに必要なグラフの作成	データをグラフにプロットできるが、適切な書式を選べていない	データをプロットしたグラフが作成できる。軸の書式等が適切に設定され必要な情報が盛り込まれている	グラフの形式を主張したい内容に応じて適切に選択でき、適切な書式でグラフを作成できる。	同上	
レポート作成の技法とワードプロセッサの利用						
	レポートを主とした文書のワードプロセッサ等での作成上の技法。論理的記述から物理的表現への変換、そのためのテンプレート、スタイル等の持つ意味。相互参照の自動処理					
		構造のある文書の作成	空白や改行の挿入など物理的表現を直接用いることで段落や見出しなどを構成できる	段落や見出しなどテンプレートの機能を用いて構成できる。 アウトラインの機能を用いて編集ができる。	レベルAに加え、必要に応じてテンプレートの調整などができる	作成した文書

		図や表の挿入	図や表を挿入した文章を作成できる。 図や表の見出し、番号を手作業で付けている	図表番号の挿入とその参照を自動化できている	レベルAに加え、必要に応じてフォーマットの調整ができる	同上
	適切な文献の書誌情報の記述とその引用、著作物の適切な引用		適切な書誌情報を付して参考文献等を挙げることができる。 剽窃と引用の違いを理解し適切な引用が行えている。	引用箇所と参考文献について同一文書内で、自動的に相互参照する機能を使える。 参考文献を適切に管理できる。	レベルAに加え、文献管理ツールなどを活用して文献の効率的な管理と活用ができる。 分野に応じた参考文献や引用の形式を理解して、それに従って文書を作成できる。	テストならびに作成した文書
	校閲など著作物を介したコミュニケーションとそのためのスキル		印刷した文書への校閲指示にしたがって文章を修正できる	校閲機能を用いて修正個所の明示化、コメントの付与、修正の反映などが行える	他者の書いた原稿に対して校閲を行える	校閲過程
ネットワークを利用したコラボレーション						
	ファイルやドキュメントの共有などグループワークを支えるネットワーク上でのコラボレーション技法		公開されているサービスを使ってファイルにアクセスできる	公開されているネットワーク上のサービスを利用して適切なアクセス制限を行い、著作権法を遵守したファイル共有が行える。	ネットワークなどを利用してグループで円滑な活動が行える	L M S の利用
	ソーシャルメディアなどの利用における注意		ソーシャルメディア上の情報の持つリスクを理解し倫理的な行動がとれる	ソーシャルメディア上の情報の持つリスクを理解し、適切な情報発信が行える	ソーシャルメディアを適切な利用が行え、積極的に利用できる	情報セキュリティ e-Learning の修了
プレゼンテーションの技法						
	スライドの作成					
		スライドの構成	スライドの内容に一部、不十分な点はあるが概ね論理的に整っている	スライドの内容が論理的に整っている	聴衆に配慮したスライドの効果的な組み立てが行えている	発表もしくはスライド
		スライドへの図や表の挿入	スライドに図や表を挿入して利用できるがプレゼンテーションでの活用が不十分である	スライドに図や表を挿入して利用できる	図や表を効果的に利用してプレゼンテーションできる	発表もしくはスライド
		視覚的効果の利用	視覚的効果を使っているが使い方が十分に適切とは言えない	視覚的効果を適切に使っている	視覚的効果を効果的に利用してプレゼンテーションできる	発表もしくはスライド
	プレゼンテーション					
		与えられた時間でのプレゼンテーション	概ね、時間を有効に活用したプレゼンテーションが出来ている	時間を有効に活用したプレゼンテーションが行える	適切な時間配分でプレゼンテーションを効果的なものにしている	発表
		質疑への対応	スライドを参照するのに手間取る、質疑を記録していない	スライドを参照しながら質疑に適切に答え、内容を記録している	質疑を想定した追加のスライドなどを準備して建設的な討議を行える	発表
		チームでのプレゼンテーション	プレゼンテーションを効果的なものにするためチームで共同作業がある程度できている。	プレゼンテーションを効果的なものにするためチームで共同作業できている。	チームのメンバーが欠けた時でも相互に補い合ってプレゼンテーションができる	発表と相互評価
	各種ダイアグラムやチャートなどグラフィカルな情報の表現技法		不十分ではあるがダイアグラムなど図的表現とそれを用いたプレゼンテーションが行える	ダイアグラムなど図的表現とそれを用いたプレゼンテーションが行える	図的表現にプレゼンテーションを効果的に行うための工夫を加えている	発表もしくはスライド
	プログラミングの基礎（オプショナル）			プログラミングを体験しているが主体的にプログラムは作れない	プログラミング言語の基本的な要素を理解しており、簡単なプログラムを書くことができる	自身の問題解決の道具として簡単なプログラミングを常に利用できる