

大学初年次におけるアルゴリズム教育

原田 融

コンピュータ・リテラシ教育からプログラミング言語へと進むための橋渡しの役割を担うものとして「BASIC」教育を位置づけ、大学初年次におけるアルゴリズム教育の試みとテキストの改訂による教育効果について報告する。教育効果の把握には毎年実施しているアンケート調査の結果を用いる。テキストに対する学生の理解度の詳細な分析は、学生に適合したアルゴリズム学習の内容と方法に重要な指針を与えることを指摘する。

1. 社会情報学部における情報処理教育

大学初年次の学生に対して、理系・文系を問わずコンピュータ・リテラシを中心とした情報処理教育が行われるようになってきている。さらに理系の場合には次の段階としてプログラミング言語教育が行われることが一般的である。札幌学院大学社会情報学部でも理系・文系の両方の学生に対して、初年次に「ワード処理」「表計算」そして「BASIC」の内容からなる「情報処理」教育を行っている。そして、2年次にはプログラミング言語として「FORTRAN」と「COBOL」を学ぶことになっている。しかし、この間に多少のギャップを感じている学生は少なくない⁽¹⁾。

本学部の「情報処理」では、学生が用意されたテキストに従って各自学習を進めていく自学自習形式を採用している。また、大学院生による教育指導員を導入することにより学生10名～12名程度に一人の割合で指導に当たる少人数教育を実現した。指導員は学生の質問に応じたり練習問題の回答をチェックする役割を担っており、自学自習用テキストと

ともに学生の習学には欠かすことのできない柱になっている。

このような教育方法のもとで初年次に開講されている「情報処理」の目的は、コンピュータになれ親しみ操作を身につけるコンピュータ・リテラシ教育だけではなく、2年次にコンピュータ言語を学ぶための橋渡しである。初年次は今後の大学4年間にコンピュータを学ぶ学生の「立ち上がり」となる大切な時期である。もしこの「立ち上がり」に失敗してしまうと、大学生活のみならず社会に出てからもコンピュータに対する苦手意識を抱いて生きるという不幸な状況にならないとも限らない。その原因の一つはキー入力などによるコンピュータの操作によることが考えられる。しかし、タイピング・ソフト等を活用すれば全くの初心者でも練習すればかなりのレベルにまで上達することができる。それゆえコンピュータを苦手と感じる主な原因は他にあると言わざるを得ない。実際、本学部の大半の学生は「ワード処理」や「表計算」をある程度マスターできても「BASIC」を習得するにはなかなか難しいと感じているようである⁽¹⁾。

我々は過去4年間にわたって実施してきた「情報処理」アンケート調査の結果に基づき、その教育内容の検討と自学自習用テキストの改訂を行ってきた。ここでは、大学初年次に行っているコンピュータ・リテラシ教育から2年次のプログラミング言語へと進むための橋渡しの役割を担うものとして「BASIC」教育を捉え、大学初年次におけるアルゴリズム教育の試みと、テキストの改訂による教育効果について報告する。

2. 「BASIC」教育の内容

教育内容の検討に入る前に、現在実施している「BASIC」教育の進め方について述べることにする。付録には今年度のテキストの内容と学習進行表を載せた。テキストでは、基本的な命令を最初に取り上げ、それぞれの例題プログラムを実行しながら、その使用規則を学生が学べるようにした(第2章)。さらに基本命令を応用した例題を提示し(第3章)、アルゴリズムに対する理解が得られるようにした。しかし、いきなりアルゴリズムを要求すると学生の理解が困難になるため、例題プログラムを理解することに重点を置き、その一部の拡張や修正で解ける練習問題を置いた。これは単なる模倣から自分で(アルゴリズムを)考える段階に向けてのステップを踏み出せることをねらいとしている。本格的にアルゴリズムを考える訓練は2年次以降のプログラミング言語に回し、ここではそのための橋渡しの効果を持つことを期待した。従って複雑なアルゴリズムを学習させることなく、さらにフローチャートやPADなどを用いた構造化プログラミングの知識は敢えて必要としないようにした。そして最も重要なことは「BASIC」教育によって「コンピュータ嫌い」をつくらないことである。

3. テキストの改訂

文献2では学生のアロリズム理解能力の

分析結果について既に報告した。そこではアルゴリズムの理解度と数学の習熟度との関連を調べ、数学的要素の強い題材においては文系の学生の理解度が低くなるということが明らかにされた。実際、学生の意見の中には「数学が苦手なので情報処理がわからない」というのが見受けられる。はたしてそれだけが原因なのであろうか? 我々の研究グループのその後の分析では、題材が数学的ではない内容に関するアルゴリズムの理解度には大きな差は生じないということが示唆されている⁽²⁾。

逆に、プログラミング言語教育がいわゆるアルゴリズムという論理的思考を効果的に訓練する場になり得るかどうか、これを模索することは今後の情報処理教育のあり方を考える一つの方向であろう。そのためには「数学がわからないからアルゴリズムが理解できない」という迷信(?)を与えないように、教育内容として取り上げる題材を厳選する必要がある。

そこで我々は、これまでのアンケート分析に加えて、レポートなどによって寄せられた意見や感想を基に、テキストの「BASIC」部分の大幅な改訂(以後、第2版)を行い、1993年度の「情報処理」から使用している。改訂した内容は主に(1)「変数」などの基本的な概念を理解し易いように詳しい説明を補充する、(2)数学的であると学生が感じる内容を削除し、日常的な生活で興味を持てる題材を取り上げる、例えば多分岐の条件判断(if~then~else~endif)を学ぶ例題として「2次方程式の解」の代わりに「ローレル指数」を扱う、(3)基本命令に関する簡単な練習問題を数多く配置し、新しく学んだ内容を理解できているか否かを自ら確認できるようにする、などである。

以下では毎年の講義(実習)最終日に実施したアンケートの分析結果からテキストの改訂による教育効果について考察しよう。アン

ケートの質問項目は実施年度によって若干の修正をしたものもあるが、原則として同じにした。そのため、対象にする学生が年度で異なってもテキストを改訂したことによる教育効果を調査することができると思われる。

具体的な考察に入る前に対象にした学生の一つの基礎データとして、1992年以降の学生の構成を図1に示す。図からわかるように高校時代の専攻が理系であった学生の割合が年々減少し、今年度は2割強程度にまでなっている。また「パソコンの実習経験がある」や「家庭にワープロを持っている」という学生は4割程度いる。にも関わらず、1, 2回ほどしかパソコンに触れたことのない学生も相変わらず約6割に達しているのが現状である。

4. テキスト内容の理解度

学生がBASICを含めた「情報処理」の内容をどの程度習得しているか、その概略を掴むために学生のテキストの内容別の理解度を見つてみることにしよう。ここでは、第1版のテキストを用いて実習を行った1992年度のアンケート結果と第2版のテキストを用いた1993年度以降のアンケート結果を比較することにした。なお1994年度のアンケートについては現在分析を行っている途中なので、その結果についての考察は次の機会に譲ること

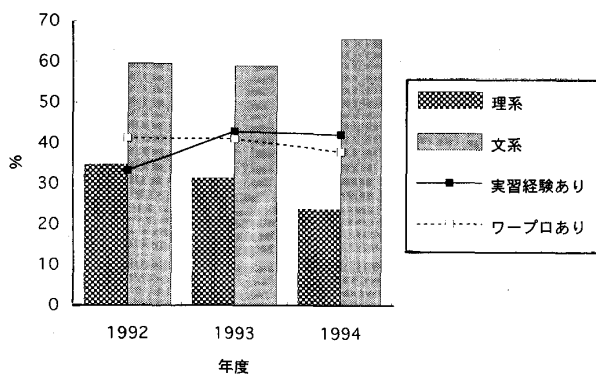


図1 初年次の学生の推移
過去3年間(1992-1994年)に入学した学生を高校時代の専攻(理系・文系)で分類した。縦軸はそれぞれの割合である。

する。

図2には「ワード処理」「表計算」および「BASIC」のテキストの各節の内容について理解し易かったかどうかを学生に尋ねた結果である。図には「よく理解できた」と「だいたい理解できた」を加えた割合を表示している。「ワード処理」と「表計算」については、最終的に7割以上の学生が理解できたと感じていることが分かる。ここでは大幅に改訂し

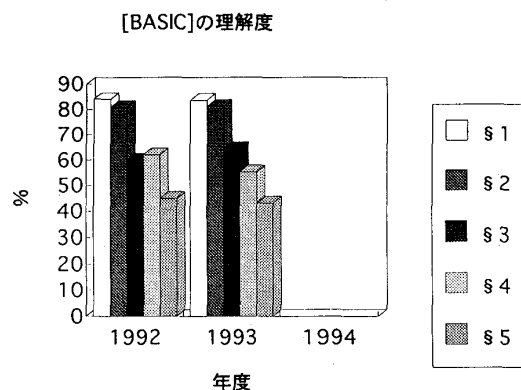
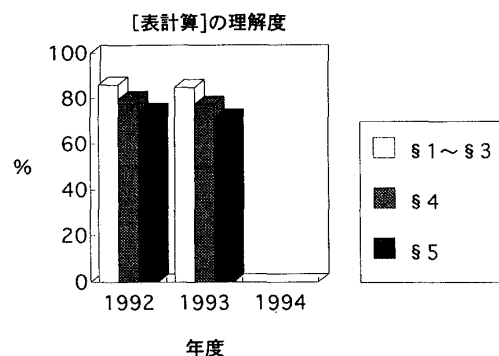
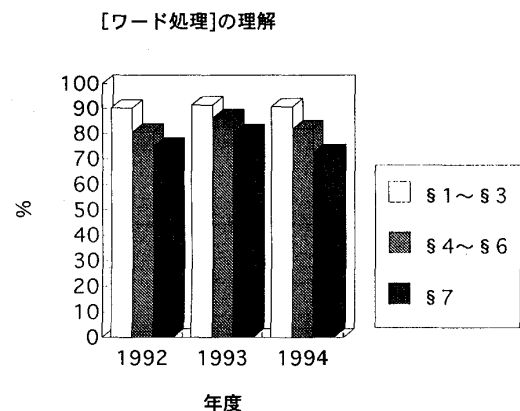


図2 テキストの内容別の理解度
「ワード処理」「表計算」「BASIC」のそれぞれの内容における各節別の理解度を示す。グラフは「よく理解できた」と「だいたい理解できた」と回答した学生の割合を加えた値を表示した。

た部分もないことから、毎年度ほぼ同じ傾向を示している。

「BASIC」に関しては第3節(基本命令の応用)と第4節(グラフィック機能の基本)の理解度に変化が見られる。1992年度の結果は文献1で分析された1991年度の結果に類似しており、理解した学生の割合が59.9%で第4節の62.1%よりも下回った結果になっている。これは2次方程式の解や三角関数などの数学的要素を含んでいることが原因と考えるとよい。一方、第2版では数学的要素を含まない題材を用いているために理解した学生の割合が増加したが、当初の期待に反して64.0%程度に留まった。この理由は、次の節で述べるように、練習問題が適切な内容として配置されているか否かに強く依存していると思われる。また第2版では第4節を理解できた学生が第1版よりも減少(55.2%)してしまった。第1版の第4節では練習問題を配置していなかったことを考えると、この理由も練習問題の扱い方が影響を与えているようである。

今回のテキストの改訂によって、より多くの学生が「BASIC」を理解できることを期待したが、アンケートの結果は最終的に理解できた学生の割合は約5割ほどで、さほど変化がみられなかった。しかしながら、次節で議論するように数学的要素を取り除いた題材のもとで得られた結果は、アルゴリズム教育の内容と方法に関して我々に重要な指針を与えていると考えられる。

5. 練習問題の理解度

テキストの各節に配置されている練習問題は、単なる例題の模倣から自分でアルゴリズムを考える段階に踏み出すための重要な役割を担っている。このテキストを自学自習することでアルゴリズムを習得させることをねらいとしているので、その題材の設定には細心の注意が必要である。しかしながら、一般に

市販されているテキストはたくさんの読者を相手にしているために、そのような点にまで気を配ることは不可能である。我々は社会情報学部初年次の学生に適合したテキストをつくるために、毎年アンケートを実施して、学生の反応を把握している。その分析をもとに教育効果の高い例題や練習問題を模索していくことが可能になると思われる。少なくとも学生のアルゴリズム習得に適した問題か否かを判定できるはずである。

そこでテキストに配置されている練習問題が理解し易かったかどうか、すべての練習問題について調べてみた。図3には第1版の練習問題の項目とその理解度の割合を示そう。数学的な要素を含んでいたために全般的に「よく理解できた」と「だいたい理解できた」を加えた割合はどの項目に対しても約5割以下である。

次に図4には第2版の練習問題についての理解度の割合を示した。ここでは基本命令に関しても数多くの練習問題を配置したために、どの段階で学生の理解が困難になっていくのかを調べることができる。特に理解度が急激に低くなる練習問題は、学生の理解のためには適していないと判断してよい。例えば「レジスター改良」「星を並べる」およびグラフィックの節にある「 $\Delta \times \Delta Y$ 」などがそれに対応している。このような練習問題は内容の説明が不十分であったり、この段階では難しすぎる設問を含んでいたことが学生の提出したレポートからも判明した。また「配列と…」など配列に関する概念は学生にとって理解するのが難しい内容のひとつである。我々はこの結果に基づいて、さらに説明を補足したり練習問題の設問の見直しを行い、1994年度のテキストとして現在使用している。今年度どのように改善されているか、現在分析中のアンケート結果が待たれるところである。

なお、ここでは学生が「練習問題を理解できた」ことで、要求された「アルゴリズムを

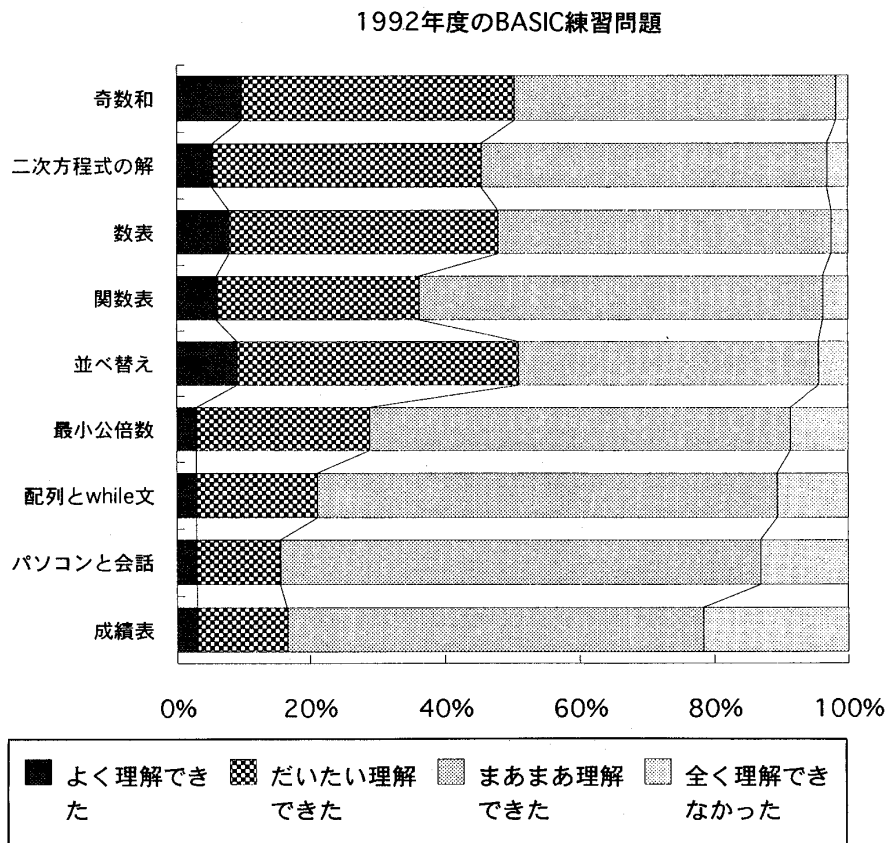


図3 1992年度(第1版)のBASIC練習問題の理解度
 それぞれの練習問題の項目別に学生の理解度を調査した。項目はテキストに配置されている順に上から表示した。

習得することができた」ものと考えている。理解できた学生の割合を増やすための有効な内容や方法を見出すためには、さらに踏み込んだ考察が必要であろう。

6. アルゴリズム習得としての情報処理教育

こうした「BASIC」教育が、大学初年次のコンピュータ・リテラシ教育から2年次のプログラミング言語への橋渡しとしてどのような効果を期待できるのであろうか。確かにBASIC言語というプログラミングを習得することも目的ではあるが、社会情報学部の学生にとってはむしろ別な役割を担うことができるように思われる。それは「基礎を積み上げて学習する」という機会を学生に提供するということである。

「情報処理」では各章の学習が終了した学生

にはワープロでA4版1枚程度のレポートの提出が義務づけられている。そのレポートを眺めていると「数学系に強い人でなければ情報処理関係を勉強するのは難しいのでしょうか」や「私は数学が苦手なので……」という書き出しの文章に出会うことが多い。彼らのように「BASIC」の学習を苦痛に感じている学生の中には数学を苦手とする人が含まれていることは少なくない。そして、このような状況を「数学ができない」ためだと思っている。アルゴリズムを理解する能力は、数学的な知識を必要としないが、どうやら数学などのいわゆる理系的教科の学習で身につけられる思考形態に関係しているようである。実際、「パズルを解いているようでとても楽しい」という感想を述べているのは理系の学生であって、そんな文系の学生はほとんどいない。

一方で「みんなよりも遅れぎみだったので

1993年度のBASIC練習問題

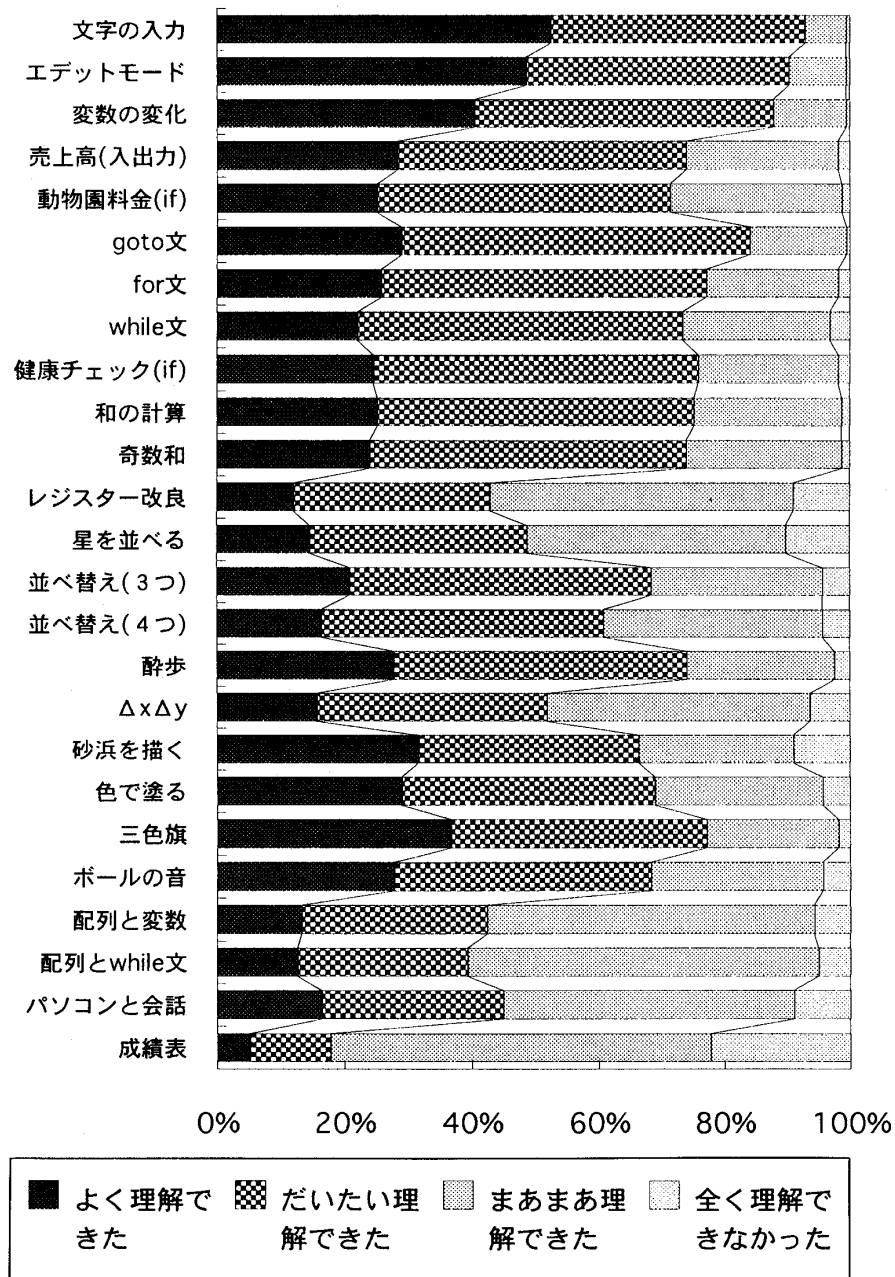


図4 1993年度(第2版)のBASIC練習問題の理解度
 それぞれの練習問題の項目別に学生の理解度を調査した。項目はテキストに配置されている順に上から表示した。

前半を飛ばしてやったら、応用に入って全く分からなくなった」「基本命令がよく分かっていないと応用でハマってしまう」「BASICは基本命令が分からないと、ぜんぜんプログラムを入力しても実行できないと分かりました」など基礎的内容をしっかり理解することの大切さを学んだという学生の意見もみられ

る。一般に学生の中には、いったい何が基礎的内容で何が応用であるかを区別して把握できていないことが多い。そして応用を理解するためには基礎的内容の理解が不可欠であることを認識していない場合も少なくない。その意味では大学初年次の「BASIC」教育が将来アルゴリズムを習得するのに必要な思考形

態を育てるきっかけとなり得ると考えられる。社会情報学部では理系・文系の如何なる学生に対してアルゴリズム習得を可能にする教育を行うことが不可欠であるために、今後の更なる具体化が必要である。

7. まとめ

大学初年次におけるアルゴリズム教育としての「BASIC」教育内容を、自学自習用テキストの改訂という見地から議論した。毎年継続してアンケートを実施し分析することによって、テキストを学生に適合した内容に近づけることがはじめて可能になった。情報処理教育はともすればコンピュータ利用環境に大きく依拠してしまい、教育内容はそれに付随するかのようになりがちである。しかし、その一方で環境に左右されることなく有効な情報処理教育のあり方を模索することが必要なのではないだろうか。

謝辞 この研究は札幌学院大学社会情報学部理系教員プロジェクトの財政的援助を受けて進められました。また「情報処理」のテキストや内容に関する議論をして頂いた森田彦氏、新國三千代氏、能登宏氏に感謝いたします。さらに数多くのご意見を寄せていただいた、佐藤将博氏をはじめ「情報処理」教育指導員の皆様、率直な感想や意見を寄せていただいた社会情報学部の学生の皆さんに感謝いたします。

文献

- (1) 森田 彦, 新國三千代: 社会情報学部における情報処理教育, 社会情報, 札幌学院大学社会情報学部紀要, Vol.1, No.2, pp.35-47(1992).
- (2) 森田 彦, 新國三千代, 原田 融: アルゴリズム理解能力の分析, 社会情報, 札幌学院大学社会情報学部紀要, Vol.2, No.2, pp.87-96(1993).

付録: 1994 年度「情報処理」実習進行予定表

担当: 原田 融・佐藤将博

<受講上の注意>

「情報処理」は、各自が自主的に学習することを基本としています(自学自習方式)。ただし、各章毎に終了期限(チェック・ポイント)を設けており、遅くともこの期限内に各章の内容を消化しなければなりません。具体的な日程は以下に示された通りです。そのためには、

- (1) テキストを必ず予習してから、講義に出席する、
- (2) 予定より遅れた場合は、予定箇所を次の講義の前までに終わらせておく、

ように努めて下さい。

なお、情報処理実習室は講義で使われていない時間帯に、オープン利用室は常時、実習のために使用することができます。また、第2・3実習室は平日夜 21 時まで使用することができます。講義時間以外に毎週、最低 1 講義時間(1 時間半)程度は実習するように心掛けて下さい。

[1] 毎回の実習に必要なものは、

- ①「情報処理」テキスト、②ライセンスディスク、③フロッピー・ディスク(5 インチ)
- [TYPEQUICK 用, OASYS 用, LOTUS 用, BASIC 用等その実習に必要なもの]です。忘れないようにして下さい。

※最初の講義で、ライセンスディスクと TYPEQUICK 用フロッピー・ディスクを各自に配布します。

[2] 講義(実習)では、各クラスに指導員(3名)が常時配置されて、皆さんの学習に適切なアドバイスを与えます。担当の指導員に、各自の進行状況のチェックを受けるようにして下さい。

[3] 「情報処理」の単位を取得するためには講義日数の 2/3 以上出席することが必要です。但し、テキストをすべて消化し学習到達目標に達した学生は、その時点で受講を終えることができます。

[前 期]

※【 】は練習問題および指導員点検内容

0. はじめに	
①講義（実習）の内容・進め方の説明	[第1回]
②「TYPEQUICK」によるキータッチの練習	04/11
I. ワード処理	
§1. 文書作成の一連の手順	
①FM-OASYSの起動 ②文書フロッピィの作成とファイル名	[第2回]
③文書作成 ④文書の題名入力 ⑤更新 ⑥印刷 ⑦終了	04/18
§2. 日本語文字の入力練習	
①一般のひらがな, カタカナ, 濁音, 半濁音, 長音, 漢字の入力方法	[第3回]
②漢字の入力方法 ③総合練習	04/25
§3. 数式・記号, 及び罫線を含む文章の入力	
①全角, 半角の英数字の入力 ②記号の入力	[第4回]
③記号を含む文章の入力練習 ④罫線の使用法	05/02
§4. 文章の編集I (基本編集機能の修得)	
①削除の仕方 ②挿入の仕方 ③移動の仕方の説明	[第5回]
④複写の仕方の説明 ⑤以前に使用した文書ファイルの使用法	05/09
§5. 文章の編集II (文章の体裁を整える機能の修得)	
①センタリング・右寄せ・左寄せ ②アンダーライン	
③倍角文字 ④改頁・頁抑止	
§6. 表の作成	[第6回]
§7. 実用的例文	05/16
①【手紙文】 ②【広告】	
③【新聞記事】 ④【講義レポート】	[第7回]
	05/23
II. 表計算	
§1. ロータス1-2-3の基本操作の概説	[第8回]
①ロータス1-2-3の起動・終了 ②フロッピィディスクの初期化	05/30
③ワークシートの説明 ④ワークシートの保存・呼び出し	
⑤ワークシートの保存(2回目以降の保存) ⑥ワークシートの印刷	
§2. ワークシートへの入力操作	
①文字・数値の入力 ②列幅の変更 ③セル内容の修正方法	[第9回]
④式の入力	06/06

§ 3. 表計算機能	
①成績表の作成 I (表作成の基本事項)	
②成績表の作成 II (データの入力)	
③成績表 III (関数の使用) ④範囲の指定 【1】	[第 10 回]
⑤罫線の引き方 【2】 ⑥関数 (③の補足) 【3】 【4】	06/13
§ 4. データベース機能	
【5】 ①範囲内検索 【6】 【7】 ②問い合わせ機能 I (基本的な使い方) 【8】 【9】	[第 11 回] 06/20
③問い合わせ機能 II (複雑な検索条件の指定) 【10】 【11】 【12】 【13】 【14】	[第 12 回]
④ソート機能 (データを並べ替える) 【15】 【16】 【17】	06/27
§ 5. グラフ機能	
①グラフ作成の方法 I (自動グラフ機能) 【18】 【19】	[第 13 回]
②グラフ作成の方法 II (データ範囲の指定) 【20】	07/04
③グラフ作成の方法 III (オプションの指定)	[第 14 回]
④XY グラフ・円グラフ・株価グラフ ⑤グラフの印刷 【21】 【22】	07/11
前期まとめ	[第 15 回] 07/18

— 夏期休暇 —

【後 期】

※ 【 】 は練習問題および指導員点検内容

III. BASIC	
§ 1. プログラム作成・実行の手順	
①フロッピーディスクの初期化 ②起動・終了 ③入力・実行	[第 1 回]
④保存・呼び出し ⑤プログラムの修正方法 【1】	09/19
⑥エディットモードの使い方 ⑦プログラムの印刷 【2】	
§ 2. BASIC の基本命令	
①PRINT 文 ②変数と代入文 【3】 ③INPUT 文 【4】	[第 2 回] 09/26
④IF 文 (条件判断) 【5】 ⑤GOTO 文 【6】	[第 3 回]
⑥FOR~NEXT 文 (回数による繰り返し) 【7】	10/03

⑦WHILE~WEND文(条件による繰り返し)【8】	[第4回]
§3. 基本命令の応用	10/17
①健康チェック【9】 ②和の計算【10】【11】	
③「九九」の表【12】 ④並べ替え(ソート)【13】【14】	[第5回]
	10/24
⑤レジスター【15】	[第6回]
	10/31
§4. グラフィック機能の基本	[第7回]
①プロローグ ①画面の座標 ②PSET文(点を打つ)【16】	11/07
③LINE文(線を引く)【17】【18】 ④CIRCLE文(円を描く)	
⑤PAINT文(色をぬる)【19】【20】	[第8回]
⑥アニメーション【21】 ⑦シミュレーション	11/14
§5. 一歩進んだBASIC命令	[第9回]
①配列(添字を持つ変数)【22】/N個のデータをソートする/	11/21
?個のデータをソートする【23】 ②READ文とDATA文	[第10回]
	11/28
③サブルーチン/パソコンと会話しよう!【24】	[第11回]
	12/05
【25】【26】	[第12回]
	12/12
後期まとめ	[第13回]
	12/19