

プログラム作成によるコンピュータ実習用教材の開発

教養部物理学 高橋 信介
taka@cc.hirosaki-u.ac.jp

理学部情報科学科 小西 栄一
konish@si.hirosaki-u.ac.jp

1 はじめに

コンピュータを利用した授業では、ここ数年間のコンピュータの目覚ましい技術開発により、その利用法の選択肢がかなり増えてきました。平成7年4月には本学の情報処理実習室の環境が、大型計算機を利用した教育システムからワークステーションやパソコンを利用した教育システムへと更新されました。情報処理実習室のさまざまな機能は日英ワープロ、表計算、画像、音声、電子メール、インターネットと進み、授業の中にもいち早く導入されています。また、いわゆる教養教育のカリキュラムも改正され、コンピュータを利用した授業はプログラミング中心のものからさまざまなアプリケーションソフトの使い方へと移りつつあります。

このような変化の中で、敢えてFORTRANによるプログラミングの教材づくりを試みました。講師が授業をおこない学生がその授業を受けるというあたりまえのことを目的に作成しました。コンピュータ初心者の受講学生が自ら考えてプログラムをつくり、そのプログラムでコンピュータを動かしてみることを学生自身に体験させていただきたい。以下にこの教材の概要と説明、サンプル、利用例、使用した感想、プログラムの項目一覧などを紹介します。

2 教材開発の概要

平成7年度総合情報処理センター研究開発費から40万円の配分を受けて教材づくりをおこないました。配分を受けた予算は情報処理実習室内のパソコン端末と同じような環境を研究室内につくるために、パソコン及びソフト(OS、通信ソフト、MS-Office等)の購入につかわれました。また、環境構築に不足するプリンタ、増設メモリ、ネットワーク接続用インターフェイス及び消耗品やコピー費等は講座費より補填しました。

担当者の担当分担は次のとおりです。サンプルプログラムの作成実行及び練習問題の追加は小西、高橋が分担しておこない、実際の授業での利用は高橋が担当し、教養部開講の情報科学(一般教育科目、自然分野)の授業で平成7年度後期におこないました。また、教材全体の配置とまとめは主に小西が担当しました。

教材研究の内容は次の通りです。これまでに担当した授業等で蓄積してきたFORTRANプログラムの例題を整理編集し、プログラミングを中心とするコンピュータ実習用教材の開発をおこないました。教材には10週分の授業の分量を盛り込み、さらに練習問題を追加しました。教材はMS-WORD ver7.0で書かれており、A4サイズで50ページ、例題、練習問題を含めて79

のプログラムから構成されています。この分量はコンピュータ初心者の学生が授業時間外に平均して授業時間の2倍程度の復習をすることで消化できることを想定しました。

3 教材の構成

3.1 概要と各回について

概要は次のような10項目から構成されています。各項目について簡単に説明します。

第1回	ワークステーションへの接続から終了まで。
第2回	テキストエディタとプログラムの実行と印刷。
第3回	練習のための例題。
第4回	変数宣言と繰り返し。
第5回	コンピュータによる条件判断と最大値の求めかた。
第6回	表記法(書式)と並び替え(大きい順)。
第7回	2次元配列変数を使って簡単な表計算をする。
第8回	偏差値を計算してみよう。
第9回	ファイルからデータを入力する。
第10回	データ検索。

3.1.1 第1回 ワークステーションへの接続から終了まで

パソコンに電源を入れるところから、電源を切るまでを説明しています。パソコン教育システムにログインして、さらにワークステーションの教育システムにログインする方法を簡単に説明しています。ワークステーションに接続してからは、パスワードの変更、mnewsによるニュースを読む、muleで電子メールを出す、mailでメールを読む、日本語の入力などを扱っています。ここで電子メールを扱うのは授業の出欠、質問及び回答に使用するためです。

3.1.2 第2回 テキストエディタとプログラムの実行と印刷

FORTRANによる四則演算を例題に、エディタのmuleを用いて、プログラムの入力、編集、保存、コンパイルと実行および印刷の方法等を実習します。プログラム例の左端に付けてある4桁の行番号は、プログラムの説明や引用のために付けたものですので、プログラムの入力の際には省くように指示して下さい。またプログラムの各行の右端に簡単な説明を付けました。

3.1.3 第3回 練習のための例題

これまでの知識だけでできる練習問題を10題用意しました。どれも単純な問題です。問題の文章のすぐ後ろに解答をつけてあります。複利による利子計算も取り上げました。問題だけを

編集して配布するのもよいし、解答も合わせて配布するのも良いと思います。あれこれ理屈を解説するよりは、はじめのうちはずぐ解答を見て慣れることが上達の方法の一つと思います。

3.1.4 第4回 変数宣言と繰り返し

配列変数と繰り返し処理を取り上げています。人間は不得意だがコンピュータが得意なものに「大量記憶」と「繰り返し処理」があります。ひとはデータを10個加算することは黙ってやるが、100個となると大人しくやる人は少ない。しかしコンピュータは10個でも、1000個でも同じであることを学生に強調していただきたい。初心者には、 $SUM = SUM + A$ という実行文がわかりづらいようです。この実行文の前と後ろで SUM の内容を表示する $WRITE$ 文を加えさせると、理解の助けになります。練習問題では級数計算、階乗計算、テーラ展開計算及び倍精度計算による数値の表現できる桁数の限界について用意しました。

3.1.5 第5回 コンピュータによる条件判断と最大値の求めかた

$IF THEN ELSE ENDIF$ を用いた条件判断による場合分けの練習です。関係演算子と論理演算子による場合分けを取り上げています。例題では、年齢を入力して20歳以上か未満かで表示するメッセージを分けています。また、この IF 文を用いて大小比較を DO ループで繰り返しおこなって、最大値を求める例題を用意しました。ここまでの知識だけでもかなりのプログラム作りを試してみることができます。他にも FORTRAN の便利なことがたくさんありますが、FORTRAN そのものを目的としていない場合はこの程度で良いのではないかと思います。もうちょっと進むために、二次元配列変数とファイルからの入力について後半に扱います。

3.1.6 第6回 表記法(書式)と並び替え(大きい順)

第5回の最大値を求めるプログラムを発展させ大きい順に並び替えるようにします。このあたりからわからなくなる学生が増えるようです。学部学科によらず同じくらいの割合で理解に時間がかかるようです。練習問題では条件に合う場合の数(例えば、成績が「優」の人数)を求める問題を加えてあります。

3.1.7 第7回 2次元配列変数を使って簡単な表計算をする

前回までに扱った1科目の50人の成績データに対し、ここで12科目50人のデータを扱うことを想定し、まず6科目5人の成績を例題とします。二次元配列変数の練習です。「I行J列のデータ」という2パラメータ変数の表現は学生には初めは苦手のようです。

3.1.8 第8回 偏差値を計算してみよう

第7回の練習問題として13題用意しました。この中の一部では、偏差値を求めるまでを平均点、分散、標準偏差及び偏差値の順に誘導する小問にわかれています。また、数学で用いる正方行列の積も扱っていますが、この解答例のプログラムにはわざと誤りを入れてあります。誤り

を入れてあることに対して不快感を強く示す学生が若干いましたが、考えようともせずただ入力する学生と、逆に誤りを見付けてやろうとする学生とのために、あちこちに入れました。単なるミスプリントのような誤りではなく、アルゴリズムを考えた上での工夫が必要です。

3.1.9 第9回 ファイルからデータを入力する

さらに大量のデータを扱うために、ファイルからデータを読み込む練習をします。12科目50人の成績のデータをあらかじめ電子メールなどで学生宛に送信しておきます。ここではファイルへの出力は省略しています。初心者は入力ファイルへ出力してしまい、必要なファイルを消してしまうことがありましたので省略しました。

3.1.10 第10回 データ検索

12科目50人の成績のデータを用いて成績処理の検索プログラムを作るために例題と練習問題を用意しました。検索の際の入力と出力ではできるだけ「対話形式」を意識してつくることが必要です。例えば、成績データのなかに無い学籍番号や科目名を検索するような場合には適切なメッセージを表示し再度入力するようにする。自分を含め使う人がどのような入力ミスをするのかを予想して、そのミスに対応できるようにすることがポイントです。

ここでは、「プログラムを作る側」と「そのプログラムを使う側」を学生に意識させていただきたい。これまでは「作った本人が使う」作業でしたが、ここでは他の学生が作った「対話形式」のプログラムをお互いに使ってもらい、あれこれと感想を言い合い、それを取り入れて対話形式の場合分けを増やす努力を期待しています。

3.2 教材のサンプル

つぎに教材の中からサンプルをいくつか紹介します。作業手順やプログラムの説明はあまり細かく書かずにメモ程度にとどめました。説明をさらに詳しく書いても学生はもっと詳しいものを要求し、それに応えるとかえって授業中の話をあまり聞かなくなり、自分でメモを取る作業をしなくなるようでしたので、説明は短くメモ程度に抑え授業中に学生がメモを書くスペースを残しました。プログラム例の左端に4桁の行番号を付けてあります。大型計算機を使ってプログラムを作ったことのある方には懐かしいものかと思います。これはプログラムの説明やプログラムの一部の引用のために付けました。プログラムの入力や実行の際にははぶいて下さい。また、プログラムの各行の右端に簡単な説明を付けました。「!」の右側はコメント部分としてコンパイラは無視しますのでそのまま残しておいてもかまいません。

3.2.1 サンプル その1

```
15) ブロック IF 文 (条件判断)、 G O T O 文
IF ( 関係式 または 論理式 ) THEN
    ( ) 内の条件が真の場合の処理 (実行文の集り)
ELSE                                     ! 偽の処理が無い
    ( ) 内の条件が偽の場合の処理 (実行文の集り) ! 時は省略可能。
ENDIF
0010C pg15-0.f ID-number, Name, Date
0030 INTEGER AGE                                     ! 整数型変数宣言
0040 WRITE(*,*) 'あなたの年齢を入力して下さい' ! 入力要求メッセージ
0050 READ(*,*) AGE                                   ! 年齢の入力
0060 IF(AGE.GE.20) THEN                             ! 大小比較 (AGE 20?)
0070     WRITE(*,*) 'You can drink 酒 !'           ! (真) の場合の処理
0080     WRITE(*,*) '                               ' ! <20 才以上>
0090 ELSE
0100     WRITE(*,*) 'You can not drink SAKE.' ! (偽) の場合の処理
0110     IA=20-AGE                                 ! <20 才未満>
0120     WRITE(*,*) 'あと',IA,'年待ちなさい。'
0130     WRITE(*,*) 'ミルクだけ飲んでなさい。'
0140     ENDIF                                     ! IF の終り
:
```

3.2.2 サンプル その2

```
16) 最大値の求めかた ( file name = pg16-0.f )
: (pg14-0.f の 0030 から 0080 と同じ) ! MAXP=62
0090     MAXP=0                                     ! I=2, P(2)=26, 26 > 62 ? No
0100     DO 100 I=1,NP                             ! 処理なし, MAXP 変らず
0110     IF( P(I).GT.MAXP ) THEN                   ! I=3, P(3)=52, 52 > 62 ? No
0120     MAXP=P(I)                                 ! 処理なし, MAXP 変らず
0130     ENDIF                                     ! I=4, P(4)=67, 67 > 62 ? Yes
0140 100 CONTINUE                                 ! MAXP=67
:
```

3.2.3 サンプル その3

17-3. 練習問題 0D-2 のプログラム (file name=pg0d-2.f) に、さらに自分の体重 (kg) も入力するようにし、健康体重との大小比較をする。体重の差が 2 (kg) 以内なら "健康" と表示し、体重差が 2 (kg) を越える場合は、超過分或いは不足分の重量を計算して表示するプログラムを追加せよ。(file name = pg17-3.f)

```
0010C    pg17-3.f    ID-number, Name ,    Date
0020C
0030  100 WRITE(*,*) '*** 健康体重による健康 CHECK ***'
0040      WRITE(*,*) '身長 (cm) を入力してください。'
0041      WRITE(*,*) '終わりたいときは 負の値を入力 '
0050      READ(*,*) H
0055      IF( H.LT.0.0 ) GO TO 9999
0060      WRITE(*,*) '体重 (kg) を入力してください。'
0070      READ(*,*) W
0075      IF( W.LT.0.0 ) GO TO 9999
0080      HW=(H-100.0)*0.9
0090      WRITE(*,*) '健康体重は',HW-2,' から',HW+2,' (kg) です'
0100      SA=W-HW
0110      IF(ABS(SA).LE.2.0) THEN
0120          WRITE(*,*) '健康です。維持してください'
0130      ELSE
0140          IF(SA.GT.2.0) THEN
0150              WRITE(*,*) SA,'(kg) 超過です。ダイエット !'
0160          ELSE
0170              WRITE(*,*) -SA,'(kg) 不足。もっとたべなさい !'
0180          ENDIF
0190      ENDIF
0200      GO TO 100
0210 9999 WRITE(*,*) 'おしまい。'
0220      STOP
0230      END
```

4 使用法の例

教養部で開講してきた一般教育科目「情報科学」の授業で使用しました。前述の通り情報処理実習室の教育システムの環境は大幅に更新されたため、それまで大型計算機 (ACOS) の教育システムで利用していた例題等をワークステーションを利用する教育システム用に修正変更して授業に用いることになりました。

4.1 授業前の準備

この教材は講師が授業をするための教材であり、学生が授業を受けるための教材であり、授業後に学生が自習するための教材です。この教材をそのまま学生に配布するだけでは済みません。まず、学生に配布するために教材の再編集が必要になります。受講する学生を念頭にいれて、毎回の授業での学生の反応を考慮しながら、プリントで配布するもの、電子メールであらかじめデータやプログラムを送信しておくもの、レポート課題にするもの等を分けて準備しておく必要があります。プログラムをメールで送信する際に完全なプログラムを送信しておく、授業での解説のあとほぼ同時に全員でそのプログラムをコンパイルと実行を行なうとコンピュータがしばらく反応しなくなり大変困ります。したがってプログラム例の一部だけを送信しておき、授業中にプリントを参考に完成させるようにしました。

4.2 授業中

授業中にパソコンの前に座っていると、パソコンに触れずにじっと解説を聞くことは学生にとって苦痛のようです。講師が解説しているときにプリントの例題を入力しはじめる学生が多いようです。全体に対しての説明はなるべく簡単に短時間にすませ、できるだけ長い時間学生がコンピュータに接するよう心がけました。あとの時間は、実習の手伝いをさせていただいていた当時の情報処理センター技術補佐員の方とともに実習室内をまわり、学生からの質問を受けて個人指導をおこないました。

はじめの1か月間に良く見掛けることが、コンピュータからのエラーメッセージが表示されていたり、トラブルが発生しコンピュータが忍し黙っていても学生もいっしょにじっとだまっていることです。気軽に質問できる授業雰囲気をつくることを心がけました。また学期末になっても時々見掛けるのですが、コンパイル時にエラーが無く、実行してエラーが出ないとそれですべてできあがりと思いついでいることです。処理されたものが正しいものかどうかの検算をしない学生が多いです。起こり得ない数値を表示していても何とも思わないでいます。

4.3 レポート課題の出題

慣れさせることを目的としてほぼ毎週のレポート提出の要求を心がけました。学期の前半のレポート課題は、解答例付き練習問題の中から指定或は複数個選択させ、後半は解答例の無いものを中心に指定しました。さらに学期末には4週間くらいかけて、大まかなテーマを出題しておいて、各自が自分で考えた処理をプログラムすることを最終レポートの課題としました。

4.4 レポートの提出法と評価

プログラムリストと実行結果をプリンタ用紙に印刷して提出させました。電子メールや教育システムの中のファイル転送も利用しました。レポート課題を出題する以上は評価することになります。受け取る度に受け取り確認の電子メールを出し、数日して評価をまた電子メールで送信します。提出されたレポートに誤りがあれば、なん度でも再提出を要求します。再提出を要求する際は誤りの場所を指摘しコメントを加えます。電子メールで書くには効率的に適切でない場合は電子メールで学生を呼び出しました。学期末にはレポート受理の可否の一覧表を授業中に回覧して再確認しました。

5 この教材を使用した感想

つぎに受講学生の声と担当者の声を紹介します。毎学期に学期の中頃と期末試験の時に感想や要望などを書いてもらいましたので、いくつかをここで紹介します。

5.1 受講学生の声

- はじめは、キーボードに触れるのも恐かった。
- はじめのうちは、全員ができるまで待っていてくれたので、大変助かりました。
- 授業が始まった頃、一番遅い人にペースを合わせて授業をすすめるのは納得がいきません。遅い人はもっと努力するべきです。ぼくはひまでした。
- なんで毎週レポート提出なのですか？ 学期末に筆記試験もやり、放課後の情報処理センターで授業時間の3倍もやって、それでたったの2単位では割りがあわない！
- 1回欠席すると次週の授業についていけない。
- 90分の授業時間が短く感じました。毎週楽しみな授業でした。
- たった1文字まちがえただけでもコンピュータは動かない。コンピュータは頭がわるい！
- プリントの説明をもっと詳しく書いて欲しい。
- 思いどおりにならないときコンピュータが壊れているのではないかと思ったことがある。
- エラーメッセージの意味がわからない。
- プログラム例に間違いを入れないで欲しい。どこかに間違いがあると思うと読む気がしない。
- 練習問題を全部やりました。おかげでキーボードをたたく速さがかなり早くなりました。
- キーボードをたたく速さだけははやくまりました。目がわるくなりました。
- 最終レポートで作った長いプログラムが完成し、おもしろ通りに動いたときは、うれしかった。
- この授業でおぼえたことは将来役にたつだろうか？

- FORTRAN は将来役に立ちますか？
- この授業で単位をもらえたら、情報処理の資格試験に合格できますか？
- ただただつらかった。これで終わりです。
- 授業時間外に情報処理実習室で質問できずに困った。
- 夜 9 時近くに、情報処理センターの実習室から帰る時、なんでこんなことをしているんだろうと思った。必修科目でもなく、教養の単位も終わっているのに。
- 放課後の情報処理センターで友達と相談しながらできたので楽しかった。
- 情報処理センターの人にいつも助けていただきました。ありがとうございました。

5.2 授業の一担当者の独り言

履修希望者が多いので週ひとコマ協力して欲しいと頼まれて引き受けた平成 2 年前期からの授業が平成 7 年後期に終了するまでの期間も含めて感想や疑問などを書きます。

- いやはやたいへんでした。いや、ほんと！
- 毎学期 300 枚前後の履修希望カードに途方に暮れた。前年度の授業内容を参考に授業の進め方を大講義室で 1 時間程説明しても希望者数の減る気配がない。止むなくその場で出席している学生の中から抽選した約 90 人に翌日までのレポート提出を要求すると 70 人くらいしか提出しない。これはどうしたことか？ さらに、そのレポートを読んでも昨日の説明を全く聞いていないと思われるものが 15 枚くらいもある。毎学期この傾向だ！ これはいったいどうしたことか？？？
- 学部指定がないためすべての学部から出席していたので、授業を進めるペースに苦労した。
- 最終レポートの作業では他学部の学生との交流も一部にありました。
- 授業時間外に授業時間の 4 倍以上の自習が必要になった学生がいた。文系理系を問わず同じ傾向があり、これでは多すぎたのだろうか？ この学生、他の教科はだいじょうぶなのだろうか？
- この学生たちが社会に出る頃は、コンピュータの環境はどんどんかわり、いまここで体験させていることはどれくらい役にたつのだろうか？ この変化に対応できることとは何だろうか？
- 授業が終盤になると、学生がつくるプログラムが 4 ページ 5 ページと増えてくると質問が多くなり、毎週がデバッグばかりになって困りはてました。私は虫取り係ではないはずなのに！
- 40 人から 50 人でスタートし学期末まで受講し単位を出した学生は 30 人くらい。上に書いた学生の感想はほぼ最後まで頑張った学生のものだが、やめて行った学生の声を聞いてみたい。
- 学生が情報処理センターの職員の方はかなり頻繁に質問をしたのにもかかわらず、職員の方が丁寧に対応していただいていたことを、学生から後に聞かされ大変感謝しております。H.M. さん、ありがとうございました。実習室には相談係の常設が必須であることを痛感しました。
- 根気よく実習を手伝って頂いた当時の情報処理センターの歴代技術補佐員のみなさんに感謝します。H.M. さん、T.N. さん、M.W. さんありがとう。この場をかりて御礼申し上げます。

6 おわりに

プログラム言語を教えるための教材というよりは、「コンピュータ初心者の学生が自分で作成したプログラムでコンピュータを動かしてみる」ための教材をつくりました。いわゆる「教養教育の授業」で使うことを想定していました。しかし、同じ教材であっても授業の目的や、授業の進め方や例題の取り上げ方によっては異なる利用法があります。試しに、この教材を理学部の数値解析物理学の授業で使ってみました。この時はこの教材の第7回目までの部分を再編集し2回に分けて配布し、ログインから FORTRAN 言語の導入までを、3回の授業でおこなうことができました。この教材がさまざまに再編集されて利用されることがあれば良いと願っています。なほ、この教材を UNIX 上に置く予定でいます。

7 付録 (プログラム例の一覧)

第2回	テキストエディタとプログラムの実行と印刷
pg08-1.f	代入、加算、出力。
pg08-2.f	入力、4則演算。
pg08-3.f	入力を促す表示する。
第3回	練習のための例題
pg0d-0.f	一辺の長さを入力し正方形の面積を計算する。
pg0d-1.f	底辺と高さを入力し三角形の面積を計算する。
pg0d-2.f	身長を入力し健康体重の計算をする。
pg0d-3.f	さまざまな数式にしたがった数値計算をする。
pg0d-4.f	半径を入力し、円周の長さ、円の面積、球の体積と表面積の計算。
pg0d-5.f	15人の成績の平均点を計算する。
pg0d-6.f	気温と湿球温度を入力し不快指数を計算する。
pg0d-7.f	三辺の長さを入力しヘロンの公式による面積の計算をする。
pg0d-8.f	二次方程式の解の計算をする。
pg0d-9.f	元金と利率を入力し複利運用の利子と元利合計の計算をする。
第4回	変数宣言と繰り返し
pg0f-0.f	配列変数の宣言と入出力と加算。
pg10-0.f	DO ループによる繰り返し加算。
pg12-0.f	DO ループによる整数型配列変数への代入と印刷。
pg12-1.f	DO ループを用いて奇数を整数型配列変数へ代入し、総和を求める。
pg12-2.f	2からはじまる2のN乗の計算。
pg12-3.f	pg12-2.fで得られた数値の逆数の和を求める。1.000に近づく。

pg12-4.f	階乗の計算をする。
pg12-5.f	指数関数のテーラ展開式の計算、自然対数の底の計算、倍精度計算。
pg12-6.f	COS 関数のテーラ展開式の計算をする。
pg12-7.f	pg0d-9.f を DO ループを用いて書き直す。
第 5 回	コンピュータによる条件判断と最大値の求めかた
pg14-0.f	DATA 文による配列変数への代入と FORMAT 文。
pg15-0.f	ブロック IF 文による条件判断。
pg16-0.f	最大値の求めかた。
pg17-0.f	最低点の求めかた。
pg17-1.f	pg12-1.f を参考にして平均点を計算する。
pg17-2.f	入力された値では三角形が出来ない場合を判断し警告を表示する。
pg17-3.f	体重と身長を入力し、健康体重からのずれを計算し、判定する。
pg17-4.f	二次方程式の重根、実根、虚根を判定し計算する。
pg17-5.f	クイズ TWO × TWO = THREE を満たす数値を求める。
pg17-55.f	pg17-5.f の別解。
第 6 回	表記法 (書式) と並び替え (大きい順)
pg18-0.f	FORMAT 文。
pg19-0.f	大きい順に並び替える。
pg1a-0.f	pg14-0.f の表示で、1 行に 15 個表示する。
pg1a-1.f	表示法の工夫をする。
pg1a-2.f	pg19-0.f に順位および得点を表示するようにプログラムを追加する。
pg1a-3.f	pg19-0.f を低い成績順に並び替えるように書き変える。
pg1a-4.f	「不可」の成績を選び出す。
pg1a-5.f	「可」の成績を選び出す。
pg1a-6.f	「良」の成績を選び出す。
pg1a-7.f	「優」の成績を選び出す。
pg1a-8-0.f	$f(x) = \sin(x)$
pg1a-8-1.f	$f(x) = \sqrt{x}$
pg1a-8-2.f	$f(x) = x^3 - 8x^2 + 11x - 8$
pg1a-8-3.f	$f(x) = \sin(x)e^{-2x}$
pg1a-8-4.f	$f(x) = \sqrt{x^2(x+1)}$
pg1a-8-5.f	$f(x) = \sqrt{x^4(1-x^2)}$
pg1a-8-6.f	$f(x) = x\sin(x)$

第 7 回	2次元配列変数を使って簡単な表計算をする
pg1b-0.f	2次元配列変数を用いて5人の6科目成績を扱う。
pg1c-0.f	6科目の平均点を計算する。
pg1d-0.f	Physicsの最高点はどれか？
pg1e-0.f	6科目の平均点の低い順に並び替える。
第 8 回	偏差値を計算してみよう
pg1f-0.f	学籍番号と科目名を指定して選び出して表示する。
pg1f-1.f	科目名を指定して、全学生の成績を表示する。
pg1f-2.f	pg1c-0.fに各科目の平均点を表示するプログラムを追加する。
pg1f-3.f	Physicsの最低点はどれか？
pg1f-4.f	各科目の最高点と最低点を求める。
pg1f-5-0.f	Physicsの平均点を求める。
pg1f-5-1.f	Physicsの分散を求める。
pg1f-5-2.f	Physicsの標準偏差を求める。
pg1f-5-3.f	学籍番号を指定してPhysicsの偏差値を求める。
pg1f-6.f	pg1f-5を参考に指定した学籍番号の6科目全ての偏差値を求める。
pg1f-7.f	全学生、全科目の偏差値を求める。
pg1f-8.f	pg14-0.fの50人の成績の順位と偏差値を求める。
pg1f-9.f	3×3の正方行列の積を求める。
第 9 回	ファイルからデータを入力する
pg20-0.f	文字を扱う。CHARACTER文
pg21-0.f	成績ファイル <code>ten</code> をオープンしてデータを入力する。
pg22-0.f	成績ファイル <code>score</code> をオープンしてデータを入力する。
第 10 回	データ検索
pg23-0.f	成績一覧から学籍番号(96Z003)と科目名(Phys)を指定し成績を検索。
pg24-0.f	対話形式を意識して、入力を促すメッセージを表示する。
pg24-1.f	再処理の問い合わせをする。
pg24-2.f	検索の対象が無い場合のメッセージ表示と処置。
pg24-3.f	2つの外国語の平均点を計算する。
pg24-4.f	最高点と最低点の学籍番号と科目名を表示する。
pg24-5.f	最高点と最低点の差が40点以上ある学籍番号と科目名をさがす。
pg24-6.f	3科目以上「不可」となったもの。
pg24-7.f	4科目以上「優」となった学生をさがす。