

C言語によるプログラミング

(情報工学実験 I - 岩村先生)

作成日 : 2005/06/30

作成者 : 松本 亮介

学部 : 工学部

学科 : 情報工学科

学籍番号 : 1030160322

§ 1、課題 1

【動作説明】

コンパイル後の実行例を記載して、分かりにくいところは補足していく。

このプログラムの動作確認をしたパソコンでは、コンパイル後の実行ファイルがすべて、`a.exe` となる。

・課題 1 - 1

-----ここから-----

```
Z:\¥test1¥岩村さんソース完成版（暫定 3 週目） ¥kadai-1>a first zwei 3 four
```

```
File name you executed is a
```

```
The NO.1 argument is first
```

```
The NO.2 argument is zwei
```

```
The NO.3 argument is 3
```

```
The NO.4 argument is four
```

-----ここまで-----

第 1 引数から順次入力されている単語を表示するプログラム。

・課題 1 - 2

-----ここから-----

```
Z:\¥test1¥岩村さんソース完成版（暫定 3 週目） ¥kadai-1>a 7
```

```
7!=5040
```

-----ここまで-----

第 1 引数に入力した数字の階上をソース上で再起的に計算するプログラム。

・課題 1 - 3

-----ここから-----

```
Z:\¥test1¥岩村さんソース完成版（暫定 3 週目） ¥kadai-1>a sample.txt
```

```
This,is,source.
```

```
File,end.
```

-----ここまで-----

第 1 引数に指定したファイルを表示するプログラム。

・課題1-4

-----ここから-----

```
Z:\¥test1¥岩村さんソース完成版（暫定3週目）¥kadai-1>dir
ドライブ Z のボリューム ラベルは CIFS.HOMEDIR です
ボリューム シリアル番号は 0000-0000 です
```

Z:\¥test1¥岩村さんソース完成版（暫定3週目）¥kadai-1 のディレクトリ

```
2006/06/30  14:41    <DIR>          .
2006/06/30  14:12    <DIR>          ..
2006/06/30  14:41                15,153 a.exe
2006/06/09  14:52                26 sample.txt
2006/06/09  13:42                328 1031060322-1-1.c
2006/06/09  13:35                444 1031060322-1-2.c
2006/06/09  14:03                578 1031060322-1-3.c
2006/06/09  14:05                770 1031060322-1-4.c
2006/06/09  14:28                893 1031060322-1-5.c
2006/06/09  14:50            1,348 1031060322-1-6.c
                8 個のファイル                19,540 バイト
                2 個のディレクトリ        101,183,488 バイトの空き領域
```

```
Z:\¥test1¥岩村さんソース完成版（暫定3週目）¥kadai-1>a sample.txt copy.txt
```

```
Z:\¥test1¥岩村さんソース完成版（暫定3週目）¥kadai-1>dir
ドライブ Z のボリューム ラベルは CIFS.HOMEDIR です
ボリューム シリアル番号は 0000-0000 です
```

Z:\¥test1¥岩村さんソース完成版（暫定3週目）¥kadai-1 のディレクトリ

```
2006/06/30  14:41    <DIR>          .
2006/06/30  14:12    <DIR>          ..
2006/06/30  14:41                15,153 a.exe
2006/06/30  14:41                26 copy.txt
2006/06/09  14:52                26 sample.txt
2006/06/09  13:42                328 1031060322-1-1.c
```

2006/06/09	13:35	444	1031060322-1-2.c
2006/06/09	14:03	578	1031060322-1-3.c
2006/06/09	14:05	770	1031060322-1-4.c
2006/06/09	14:28	893	1031060322-1-5.c
2006/06/09	14:50	1,348	1031060322-1-6.c
		9 個のファイル	19,566 バイト
		2 個のディレクトリ	101,183,488 バイトの空き領域

-----ここまで-----

第 1 引数にもとから存在するファイルを指定し、第 2 引数で指定したファイル名で、同じフォルダ内に第 1 引数で指定したファイルと同じ内容のファイルを作成するプログラム。

上記の場合、sample.txt と copy.txt は同じ内容のファイルである。

・ 課題 1 - 5

-----ここから-----

Z:¥test1¥岩村さんソース完成版（暫定 3 週目）¥kadai-1>a sample.txt

-- Line 1 --

Field 1: This

Field 2: is

Field 3: source.

-- Line 2 --

Field 1: File

Field 2: end.

-----ここまで-----

引数のファイルに含まれる文字列を、カンマを区切りとして、行数と単語数に整理して表示するプログラム。

この場合の sample.txt のファイルの中身は以下。

-----ここから-----

This,is,source.

File,end.

-----ここまで-----

・課題 1 - 6

-----ここから-----

```
Z:¥test1¥岩村さんソース完成版（暫定 3 週目） ¥kadai-1>a sample.txt 2 3
```

```
-- Line 1 --
```

```
Field 1: This
```

```
Field 2: source.
```

```
Field 3: is
```

```
-- Line 2 --
```

```
Field 1: File
```

```
Field 2: end.
```

-----ここまで-----

課題 1 - 5 のように、第 1 引数のファイルに含まれる文字列を、カンマを区切りとして分け、第 2 引数と第 3 引数で指定されたフィールド数の単語を入れ替えて表示するプログラム。

ただし、入れ替えるように指定されたフィールドの一方もしくは両方が存在しなかった場合は、入れかえを行わない。

この場合の `sample.txt` のファイルの中身は以下。

-----ここから-----

```
This,is,source.
```

```
File,end.
```

-----ここまで-----

§ 2、課題 2

【インストール方法】

1、1031060322-2-1.c をコマンドプロンプトにより、コンパイルする。

アクセサリから、コマンドプロンプトを開き、コンパイラのパスが通っているフォルダに、1031060322-2-1.c を移動し、コマンドプロンプトに、コンパイラが `gcc` の場合は、

```
gcc 1031060322-2-1.c
```

と入力して、**ENTER** を押す。

2、次に、コンパイルによってできた実行ファイル（それぞれの環境にもよるが、自分の場合は **a.exe** というファイル。一般的には、**1031060322-2-1.c** というファイルができると思われる。）をコマンドプロンプト上から実行する。

コマンドプロンプトに入力するのは、データベースファイルを第一引数にとって入力する。データベースファイルが、**student.txt** の場合は、

```
a.exe student.txt
```

と入力する。

これにより、本プログラムが実行される。

3、データベースファイル作成ルール

ここでデータベースである、**student.txt** のデータベース作成のルールを説明する。

まずは、実際にデータベースに入力してある例を載せておく。

-----ここから-----

```
145614,Koichi Fujita,MUSIC
145615,HItoishi Mori,READING
145616,Yutaka Mizuta,PROGRAMING
145617,Ryousuke Matsumoto,PC
145618,Saki Moriyasu,SPORTS
145619,Yuichi Tasaka,GAME
145620,Cherinn Sonn,SOCCKER
145621,Takeshi Nomura,VIDEO
145622,Yuichiro Okuda,RUGBY
145613,Yasuhisa Ikuta,RUGBY
145612,Yshiharu Tuzaki,SOCCKER
145623,Hirohisa Kanou,SOCCKER
145611,Emi Tamaoka,SHOPPING
```

-----ここまで-----

上記のように、データベースを入力して、テキストファイルで保存していく。

本プログラムを実行し、その中でデータベースを記録していくことも可能（下記で説明）であるが、ここではテキストファイルとして手動で入力する方法を説明する。

上記の例のように、1行にその生徒のデータを入力する。

[学籍番号],[名前],[趣味]

という順番で入力し、一人のデータの入力が終わったら、改行して次の行に別の人のデータを順次入力していく。

データの入力が終われば保存して、プログラムの実行ファイルがある同じフォルダ内においておく。

【使用方法】

1、上記のインストール方法により実行すると、以下のような画面になる。

-----ここから-----

[search]: search data
[number]: search data from number
[name] : search data from name
[hobby] : search data from hobby
[check] : check all data
[add] : add data
[save] : save data
[finish]: finish searching data

Enter your question :

-----ここまで-----

2、データベース検索方法。

"Enter your question :"に対し、上に表示されている8つの項目の中から、検索方法を選び入力する。

ここでは、ひとつずつ説明していく。

I、 search -- 入力すると、以下のような質問が出力される。

-----ここから-----

Enter your question :search

Enter number & question[example - '145616 name']:

-----ここまで-----

ここで、:以下に、example であげられているように、学籍番号と調べる種類をスペースをあけて入力する。

example の場合でいくと、145616 の学籍番号の名前を調べたい、ということになる。

name の他に hobby を入力することもできる。

実際に入力した例を見てみると、以下となる。

-----ここから-----

Enter your question :search

Enter number & question[example - '145616 name']:145616 name

The name of No.145616 is 'Yutaka Mizuta'

press enter key...

-----ここまで-----

-----ここから-----

Enter your question :search

Enter number & question[example - '145616 name']:145616 hobby

The hobby of No.145616 is 'PROGRAMING'

press enter key...

-----ここまで-----

II、 number -- 入力すると以下のような質問が出力される。

-----ここから-----

Enter your question :number

Enter number:

-----ここまで-----

ここで、: 以下に学籍番号を入力することで、その学籍番号の生徒のデータを出力することができる。

実際に入力した例をみると、以下となる。

-----ここから-----

Enter your question :number

Enter number:145619

-----DATA OF 145619-----

NUMBER:145619

NAME :Yuichi Tasaka

HOBBY :GAME

press enter key...

-----ここまで-----

III、name -- 入力すると以下のような質問が出力される。

-----ここから-----

Enter your question :name

Enter name:

-----ここまで-----

ここで、:以下に名前を入力することで、その名前の生徒のデータを出力することができる。
実際に入力した例をみると、以下となる。

-----ここから-----

Enter your question :name

Enter name:Yuichi Tasaka

-----DATA OF 145619-----

NUMBER:145619

NAME :Yuichi Tasaka

HOBBY :GAME

press enter key...

-----ここまで-----

IV、hobby -- 入力すると以下のような質問が出力される。

-----ここから-----

Enter your question :hobby

Enter hobby:

-----ここまで-----

ここで、:以下に趣味を入力することで、その趣味に該当する生徒のデータを出力することができる。

実際に入力した例をみると、以下となる。

-----ここから-----

Enter your question :hobby

Enter hobby:SOCCER

-----DATA OF SOCCER-----

NUMBER:145620

NAME :Cherinn Sonn

HOBBY :SOCCER

-----DATA OF SOCCER-----

NUMBER:145612

NAME :Yshiharu Tuzaki

HOBBY :SOCCER

-----DATA OF SOCCER-----

NUMBER:145623

NAME :Hirohisa Kanou

HOBBY :SOCCER

press enter key...

-----ここまで-----

V、check -- 入力すると以下のように、データベースファイルのすべてのデータを確認することができる。

(後述する add で追加したデータも、save をしていなくても、その追加したときの状態で、ソフトを終了しない限りは、check で確認することができる。)

-----ここから-----

Enter your question :check

-----DATABASE-----

145614,Koichi Fujita,MUSIC

145615,Hitoshi Mori,READING

145616,Yutaka Mizuta,PROGRAMING
145617,Ryousuke Matsumoto,PC
145618,Saki Moriyasu,SPORTS
145619,Yuichi Tasaka,GAME
145620,Cherinn Sonn,SOCCKER
145621,Takeshi Nomura,VIDEO
145622,Yuichiro Okuda,RUGBY
145613,Yasuhisa Ikuta,RUGBY
145612,Yshiharu Tuzaki,SOCCKER
145623,Hirohisa Kanou,SOCCKER
145611,Emi Tamaoka,SHOPPING

press enter key...

-----ここまで-----

VI、 add -- 入力すると以下のような質問が出力される。

-----ここから-----

Enter your question :add

Enter number:145624

Enter name:Migiwa Sanaka

Enter hobby:SHOPPING

press enter key...

-----ここまで-----

順次、number、name、hobby の順で入力する場所が現れるので、それにしたがって入力し、最後に ENTER を押すことで、追加を完了する。

ただし、これは配列に格納したに過ぎず、データベースファイル自体を書き換えたわけではないので注意する。

VII、 save -- 入力することで、add で追加したデータを、データベースファイルに記録することができる。

save するには、以下のような質問が出力される。

```
-----ここから-----  
Enter your question :save  
SAVE?(y/n):y  
Save complete!!  
press enter key...  
-----ここまで-----
```

2行目で、yかnを入力することで、ファイルに変更を記憶することができる。

VIII、finish -- 入力すると、このソフトを終了することができる。

6.2 検討

1、プログラムの更なる工夫について

大体のことはできるようなプログラムにした。

更なる工夫を加えるとすると、

- ・データベースの種類を3つから増やす
- ・エラー出力時のその時に適当なエラー処理を施す

等が挙げられる。

2、工夫に対するアルゴリズム

データベースの種類を増やすアルゴリズムは、簡単にできそうである。

- ・構造体の種類を3つから増やす
- ・趣味や名前を探するときと同じアルゴリズムを同様にそれらに施す

これだけでできそうな気がした。

エラー出力時の適当なエラー処理は、意外と手間がかかりそうなものが多いと思った。

大体のエラー処理は、continue でスキップするようにしたが、そうしたのにはループの中に

エラー出力の構成部分が組み込まれているため、簡単に `printf` でエラー表示や、再度入力画面等にするには、そのループの構成自体を少し変えないといけなかった。

これは、今回少し課題の残る内容だとおもった。

時間があれば、どのようにループとエラー部分を分けて構成するかを考えたいと思う。

6.3 実験の感想

今回のデータベースプログラムは、考えていた構成通りに組むことができたと思う。

今まで組んできたプログラムは、練習的なものが多かったが、今回取り組んだプログラムは「ソフト」といったような感じがあって、組んでいくほどにその感覚が強くなり、できたときは達成感があった。

このような「ソフト」をプログラムでこれからもどんどん組んでいきたいとも思え、この実験でプログラムを組むことができ、自分のためになったとも思えた。

夏は、暇があればこのプログラムをもっと改良して、C++等で GUI を取り込んで組むのもおもしろいかなーとおもった。