

プログラミング演習 1

第 1 回レポート

氏名: 池田 海斗 (IKEDA, Kaito)
学生番号: 09501502

出題日: 2020 年 04 月 22 日

提出日: 2020 年 04 月 22 日

締切日: 2020 年 04 月 30 日

※執筆上の注意：本書は空想上の課題に対するレポートの執筆例である．章の構成と書くべき内容の参考として提示するものであるため，課題内容やプログラムの仕様などは，実際の演習課題の指示に従って適切にまとめ直す必要がある．文や文の一部を「・・・」や「??」によって省略している箇所があるが，これに穴埋めをするだけで，レポートが完成するわけではない．なお，サンプルと同じ書き出しで文章を書く必要もない．

1 概要

※執筆上の注意：概要は多すぎず少なすぎずが重要である．特に，次の 3 点について，執筆者の取り組みの概略が読者（＝教員）に伝わるようにしよう．(1) このレポートで取り組んだ課題の内容，(2) 実験等によって得られた結果，(3) 結果に対しておこなった考察．

本演習では，名簿管理機能を有するプログラムを，C 言語で作成する．このプログラムは，標準入力から「ID, 氏名, 誕生日, 住所, 備考」からなるコンマ区切り形式（CSV 形式）の名簿データを受け付けて，それらをメモリ中に登録する機能を持つ．ただし，% で始まる入力行はコマンド入力と解釈し，登録してあるデータを表示したり整列したりする機能も持つ．

本レポートでは，演習中に取り組んだ課題として，以下の課題 1 から課題 3 についての内容を報告する．

課題 1 文字列操作の基礎：subst 関数と split 関数の実装

課題 2 構造体や配列を用いた名簿データの定義

課題 3 標準入力の・・・（サンプルのため，省略）

また，取り組んだ課題のうち，特に以下の課題については，詳細な考察を行った．

課題 1 文字列操作の基礎：subst 関数と split 関数の実装

課題 3 標準入力の・・・（サンプルのため，省略）

2 プログラムの作成方針

※執筆上の注意：追加仕様については，書かなくても良い．書く場合は，次節の説明などでも，その仕様を参照しながら書くといよい．

本演習で作成したプログラムが満たすべき要件と仕様として，「(1) 基本要件」と「(2) 基本仕様」を示す．

(1) 基本要件

- 1. プログラムは、その実行中、少なくとも 10,000 件の名簿データをメモリ中に保持できるようにすること。
- 2. 名簿データは、・・・
- 3. プログラムとしての動作や名簿データの管理のために、・・・
 - (a) プログラムの正常な終了
 - (b) 登録された・・・
- 4. 標準入力からのユーザ入力を通して,,,

(2) 基本仕様

- 1. 名簿データは、コンマ区切りの文字列（CSV 入力と呼ぶ）で表されるものとし、図 1 に示したようなテキストデータを処理できるようにする。
- 2. コマンドは、% で始まる文字列（コマンド入力と呼ぶ）とし、表 1 にあげたコマンドをすべて実装する
- 3. 1 つの名簿データは、C 言語の構造体 (struct) を用いて、・・・

※執筆上の注意：繰り返すが、本書は空想上の課題に対するレポートの執筆例である。書くべき内容としては、課題書等を読んだうえで適切に執筆すること

3 プログラムの説明

※執筆上の注意：変数や数値は \verb や \$\$ で囲って、適切な書体で記述することを忘れずに。なお、このサンプルでは“わざと”一部の処理を省略している。見た目の違いを確認して、自分のレポートでは処理を忘れないようにしよう。また、この章はこのレポートサンプルの 2 章に基づいて書かれているが、そもそも 2 章が架空の講義内容に基づいて書かれている点に注意すること。

プログラムリストは・・・節に添付している。プログラムは全部で 267 行からなる。以下では、1 節の課題ごとに、プログラムの主な構造について説明する。

1:	5100046,The Bridge,1845-11-2,14 Seafield Road Longman Inverness,SEN Unit 2.0 Open
2:	5100127,Bower Primary School,1908-1-19,Bowermadden Bower Caithness,01955 641225 ...
3:	5100224,Canisbay Primary School,1928-7-5,Canisbay Wick,01955 611337 Primary 56 3...
4:	5100321,Castletown Primary School,1913-11-4,Castletown Thurso,01847 821256 01847...

図 1 名簿データの CSV 入力形式の例。1 行におさまらないデータは... で省略した。

表 1 実装するコマンド

コマンド	意味	備考
%Q	終了 (Quit)	
%P n	先頭から n 件表示 (Print)	n が 0 → 全件表示, n が負 → 後ろから -n 件表示
*****	(サンプルのため 省略)	

3.1 文字列操作の基礎：subst 関数と split 関数の実装

まず、汎用的な文字列操作関数として、`subst()` 関数を 11-25 行目で宣言し、`split()` 関数を 27-44 行目で宣言している。また、これらの関数で利用するために、・・・(省略)

`subst(STR, C1, C2)` 関数は、`STR` が指す文字列中の、文字 `C1` を文字 `C2` に置き換える。プログラム中では、`xxx` 関数の中で、入力文字列中の末尾に付く改行文字をヌル文字で置き換えるために使用している。

`split(STR, ...)` 関数は、・・・(※サンプルのため、いろいろと省略)

3.2 構造体や配列を用いた名簿データの定義

本名簿管理プログラムでは、構造体の配列を名簿データとして扱う。18-27 行目で、`date` 構造体を定義し、29-48 行目で、`profile` 構造体を定義している。この・・・が、名簿データ 1 つに相当する。そして、`xxx` 行目の `xxxxxx` 変数で、全名簿データを管理し、`xxx` 行目の `xxxxxx` 変数で、名簿データの個数を管理する。

`date` 構造体の定義にあたっては、・・・(以降、サンプルのため、省略)

・・・(サンプルのため、省略)

3.3 標準入力の・・・(サンプルのため、省略)

・・・(サンプルのため、省略)

4 プログラムの使用法と実行結果

※執筆上の注意：この節はプログラムの使用法を説明する節である。最低限、起動の方法、入力の形式と方法、出力の読み方を入れること。当然、実装したコマンドすべてを説明すべきであるが、このサンプルのように説明に使う実行例が 1 つである必要はない。適宜、subsection などを用いて見やすく整理するとよい。なお、本学科の演習室では Red Hat Linux は利用していない。

本プログラムは名簿データを管理するためのプログラムである。CSV 形式のデータと % で始まるコマンドを標準入力から受け付け、処理結果を標準出力に出力する。入力形式の詳細については、??節で説明した。

プログラムは、Red Hat Linux 3.2.2-5 で動作を確認しているが、一般的な UNIX で動作することを意図している。なお、以降の実行例における、行頭の\$記号は、Red Hat Linux 3.2.2-5 におけるターミナルのプロンプトである。

まず、`gcc` でコンパイルすることで、プログラムの実行ファイルを生成する。ここで、`-Wall` とは... するためのオプションであり、`-o` とは・・・を指定するオプションである。これらのオプションをつけることで、・・・(略)

```
$ gcc -Wall -o program1 program1.c
```

次に、プログラムを実行する。以下の実行例は、プログラム実行中の・・・を模擬するため、・・・を・・・により与えることで、実行する例を示している。通常の利用においては、・・・(略)

```
$ ./program1 < csvdata.csv
```

プログラムの出力結果として、CSV データの各項目が読みやすい形式で出力される。例えば、下記の `csvdata.csv` に対して、

```
0,Takahashi Kazuyuki,1977-04-27,3,Saitama,184,78
10,Honma Mitsuru,1972-08-25,2,Hokkaidou,180,78
```

%S3
%P0

以下のような出力が得られる.

```
code: 10
name:  Honma Mitsuru
bday:  1972/08/25
type:  2
home:  Hokkaidou
height: 180
weight: 78

code: 0
name:  Takahashi Kazuyuki
bday:  1977/04/27
type:  3
home:  Saitama
height: 184
weight: 78
```

まず, 入力データについて説明する. 入力中の最初の 2 行で, 2 つの CSV データを登録している. CSV データは, 表 1 で示したように, 7 つの項目からなる. 3 行目の %S3 は, これまでの入力データを 3 番目の項目 (生年月日) でソートすることを示している. 4 行目の %P0 は, 入力した項目の全ての項目 (1-7) を表示することを示している.

(※サンプルのため省略)

※執筆上の注意: 良くないレポートの書き方を 2 つ挙げる.

- (1) 入力の説明が無いため, 何をしたのかがわからない.
- (2) 出力結果だけが書いてあり, それが「想定 (仕様) 通り」なのかどうか, を述べていない.

5 考察

※執筆上の注意: 考察課題を中心にまとめる. あくまで「作成したプログラムとその実行結果から考察できること」を, 主に求めている. 単なる感想で終わるような内容を書いてはいけない.

・・・章のプログラムの説明, および, ...章の使用法と実行結果から, 演習課題として作成したプログラムが, [?] 章で述べた基本要件と基本仕様のいずれも満たしていることを示した. ここでは, 個別の課題のうち, 以下の 3 つの項目について, 考察を述べる.

1. 文字列操作の基礎: subst 関数と split 関数の実装
2. 標準入力の・・・(サンプルのため, 省略)
3. ... (サンプルのため, 省略)

5.1 「文字列操作の基礎: subst 関数と split 関数の実装」に関する考察

(※サンプルのため省略)

5.2 「標準入力の・・・(サンプルのため、省略)」に関する考察

(※サンプルのため中略)

5.2.1 「・・・(サンプルのため、省略)」に関する考察

(※サンプルのため省略)

6 感想

(※サンプルのため省略)

7 作成したプログラム

※執筆上の注意：はみ出さないように注意。また，??章のような状態でレポートを提出しないように注意すること。

作成したプログラムを以下に添付する。なお，1 章に示した課題については，??章で示したようにすべて正常に動作したことを付記しておく。

```
1: #include <stdio.h>
2:
3: int main()
4: {
5:     char s[4] = {'A', 'B', 'C', '\0'};
6:
7:     printf("s = %s\n", s);
8:
9:     return 0;
10: }
```

※執筆上の注意：余白部分に文字がはみ出していないか，よく確認する。例えば， \LaTeX によるコンパイル時の Warning メッセージを確認しよう。Overfull hbox が出ていたら，はみ出している場所があるはずである。

参考文献

- [1] 平田富雄，アルゴリズムとデータ構造，森北出版，1990.
- [2] 著者名，書名，出版社，発行年.
- [3] WWW ページタイトル，URL，アクセス日.

※執筆上の注意：これらは書き方の例である。実際に，参考にした書籍等を見て書くこと。