

# プログラム課題のいくつか

2002/12/20 Ayahiko Niimi

C 言語の復習用のプログラム課題、アルゴリズムとデータ構造に関する課題、ちょっと大きめなプログラムの課題、パズル課題を作りました。

どれでもかまいませんので、5個ほど解いてみてください。一応、いくつかはヒントを載せてあります。それぞれの難しさは、まちまちですし、ヒントの詳しさもまちまちです。

一応、C 言語を想定して作りましたが、他の言語でもかまいません。自分の勉強したい言語、ポイント、設計の仕方などを考えながらやってみてください。

どこかにある、プログラムや解説を参考にしてもかまいません。

とにかく、数をこなしてください。「自分でたくさんプログラムを書く」、「人のプログラムをたくさん読む」がプログラム上達の近道です。

(課題の中に含まれていないテーマ)

- GUI
- 画像処理
- 数値計算
- データベース
- ネットワーク/通信
- プロセス管理
- ...

## 今あるプログラムの作成(あるいは車輪の再発明)

### UNIX コマンド

UNIX の代表的なコマンドを自分で作ってみてください。問題を簡単にするために、ファイルの中身は、半角英数のみとします。

- cat: ファイルの表示
- tac: ファイルを末尾から表示
- head: ファイルの先頭10行を表示
- tail: ファイルの末尾10行を表示
- sort: ファイルをアスキーコードの順に並びかえる。
- sort-n: ファイルを数字の順に並びかえる。
- wc: ファイルの文字数、単語数、行数を数える
- fold: ファイルを 1 行80文字で折り返す。

### C 言語の標準ライブラリの作成

C 言語のライブラリ関数を作ってみてください。

- abs: 絶対値を返す
- strcat: 文字列の連結
- strcmp: 文字列の比較
- strcpy: 文字列のコピー
- strlen: 文字列の長さを返す
- strtok: 文字列を区切る
- atoi: 文字を数字に変える

### おまけ

- strup:小文字を大文字に変換して返す
- strlow:大文字を小文字に変換して返す

さらに余力があれば、

- log
- sin
- cos

### 行列計算ライブラリ

行列計算用のライブラリを作ってください。

- 足し算
- 引き算
- かけ算
- 逆行列
- 固有値
- 固有ベクトル

## ソートプログラムの評価

1-1000までの整数からランダムに1000出力するプログラムを作る

バブルソートアルゴリズムによるソートプログラムを作る

配列の後ろから先頭に向かってスキャンしていき、もし隣り合う2つの要素の大小関係が逆だったら、それを入れかえる。

ビンソートアルゴリズムによるソートプログラムを作る

要素のとりうる値に対応する配列を用意する。整列する配列の要素を順番にとり出し、対応する配列を埋めていく。全部取り出し終わったら、対応する配列のうち、埋まっている部分だけを順番に読み出す。

分布数え上げソートアルゴリズムによるソートプログラムを作る

すべての要素をスキャンして、要素の出現頻度を調べる。出現頻度の累計から要素の出現位置を調べる。

選択ソートアルゴリズムによるソートプログラムを作る

整列されていない部分から最小の要素を選び出して、それを先頭に持っていく。

挿入ソートアルゴリズムによるソートプログラムを作る

配列のうち一部分を整列済みの状態にしておき、残りの要素を

1 つずつその中の適切な位置に挿入していく。

クイックソートアルゴリズムによるソートプログラムを作る  
配列から基準の要素を選び出し、配列の要素を1 つずつ基準の要素と比べる。基準の要素より大きいものは基準要素の右側に、小さいものは左側に移動する。すべての要素に対して比較が終わったら、基準要素の左側にある要素と右側にある要素に対して、同じことを行う。それ以上分割できなくなったとき、終了する。

二分探索木によるソートプログラムを作る

任意の節 x について、左部分木に含まれる要素は節 x よりも小さく、右部分木に含まれる要素は節 x よりも大きい  
(c.f.完全二分木：根からすべての葉までの経路の長さが等しい、平衡木：すべての節において、左部分木と右部分木の高さの差が1以内に収まる)

どのソートが一番速いですか

### 実用(?)プログラム

逆ポーランド電卓を作る

1 2 + 3 4 - \* =

(1+2) \* (3-4) =

(ヒント)

スタックを使うと実現できます。

元のテキストをランダムに切り貼りするプログラム  
切り貼り対象ファイルを、適当に裁断し、適当に組みあげるプログラムを作ってください。

単語出現頻度

入力したファイルの中の単語を出現頻度順に並べるプログラムを作ってください。

出力例

---  
281 chaos  
267 system  
249 of  
244 method  
239 neural  
239 network  
238 circuit  
...  
---

略語辞書プログラム

略語を調べられる辞書プログラムを作ってください。  
入力した略語が初めてなら、辞書に登録する。すでに入力された略語なら、辞書に追加する。

辞書の一例

---

FUN,Future University-Hakodate  
CD,Compact Disk  
WYSIWYG,What You See Is What You Get  
LSI,large-scale integration  
NGO,nongovernmental organization

---

ちょいメモソフトを作ってください。  
メモを簡単に取れるソフトを作ってください。

例として、個人的趣味から、コマンドラインの引数をメモの内容とするプログラム考えてみました。コマンド名が m だとして、次のような動作を考えました。

(実行例)

% m sample  
% m hoehoe  
% m hoehoe hogehoge

(メモの中身)

% cat memo.txt  
2001/04/24 (Tue) 01:47:25  
sample

2001/04/24 (Tue) 02:16:31

hoehoe

2001/04/24 (Tue) 02:17:59

hoehoe hogehoge

メモファイルの書式は

<メモソフトを起動した日時>

<実行時の引数の内容>

<空行>

の繰り返しです。

メモファイルのファイル名はとりあえず memo.txt で固定にしました。

# 毎回、ファイル名を指定するのも面倒なので。

メールの中から、プログラムソース部分を抜き出すプログラム

メール中に ">-- begin cut" で始まる行から ">-- end cut" で始まる行までを取り出すプログラムを作ってください。

気合いがある人は、以下の機能を足してみては?

- ファイル名をタグから取り込む
- ファイルの更新日時をタグから取り込む
- 逆にメールに埋め込むためのタグをつける

ファイルを安全に削除できるように、ファイルの中身の文字をすべて "\*" で置き換える、前処理プログラムを作ってください。

ファイルの中身を書き換えてしまうので、ファイル復元関係の

ツールでも復元しにくくなります。

(ヒントその1)

まず、cat プログラムを作ってみてください。  
ファイルを読み込んで、そのまま表示するプログラムです。  
# for, scanf, printf などをつかえば、簡単に書けるとおもいます。

(ヒントその2)

ヒント1のプログラムの printf の辺りを変更して、読み込んだ文字の分だけ、"\*"を表示するように変更してください。文字列の長さをチェックする関数(メソッド)には、以下のようなものがあります。(詳しくは、文法書をチェックしてください。)

```
- size_t strlen(const char *s); /* for Lang C in string.h */
- public int length() // for Java in java.lang.string
```

(ヒントその3)

ヒントではないのですが、これでは簡単すぎるという人は、ファイルの削除までプログラミングしてみてください。  
# これでも簡単だという強者は、ファイルではなく、ディレクトリを指定したときの動作を考えてみてください。

## パズル

8クイーン問題を解くプログラムを作ってください。  
チェス盤の上にクイーンを8個置いてください。  
5x5の大きさの魔法陣を作るプログラムを作ってください。  
5x5のマスに1から25までの数字を置いて、魔法陣を作ってください。

# 魔法陣とは、各縦、横、斜めの列の和が等しいものです。

小町算

1から9間での数字を順に並べて、+,-を補って100を作る組み合わせを考えてください。

(例)

123 - 45 + 67 + 89 = 100

小銭の払い方

n円をk円以下の小銭で払う方法を教えてください。

(例)

12円を払うなら

- 10円玉1枚と1円玉2枚
- 5円玉2枚と1円玉2枚
- 5円玉1枚と1円玉7枚
- 1円玉12枚

実行した結果が、もとのソースプログラムになるプログラムを作ってください。

実行した結果が、もとのソースプログラムになるプログラムを作ってください。

例) source.c --コンパイル--> a.out --実行結果--> output.c

output.cとsource.cが同じになればよい。

コンピュータと数当てゲーム

ゲームの大まかな手順を以下に示します。

# コンピュータとの対戦ゲームです。

- 1) コンピュータが乱数によって4桁の数を決める。(ユーザ側には隠しておきます。)
- 2) ユーザがコンピュータの決めた数を推理してキーボードから4桁の数を入力する。
- 3) ユーザが入力した数とコンピュータが決めた数を比べて、コンピュータがヒントを出力する。
- 4) コンピュータが出力したヒントを元に、もう一度コンピュータの数を推理して、キーボードから4桁の数を入力する。
- 5) 3)と4)をコンピュータの決めた数とユーザが入力した数が等しくなるまで繰り返す。
- 6) なるべく、少ない回数で推理が当たるようにする。

3) のヒントの出し方を2通り考えました。

[a] ユーザの数とコンピュータの数の大小をヒントにする

- "等しい",
- "コンピュータの数の方が大きい",
- "コンピュータの数の方が小さい"

# 例

computer -> 1932

input -> 1234

hint -> もっと大きいです

[b] ユーザの入力した数とコンピュータの決めた数の中で"何桁が正しい数か"と使われている数字はいくつあっているか"をヒントにする。

- "ユーザの入力した数字は?個が正しい桁に正しい数字が使われています"
- "ユーザの入力した数の中で?個がコンピュータの決めた数字の中に使われています"

# 例

computer -> 1932

input -> 1234

hint -> 正しい桁の数は2個です。正しい数字は3個です。

・ (その1) 数の大小による判断。(パターン[a])

"EQUAL", "BIGGER", "SMALLER"のヒントを元に、コンピュータの考えた数字をあてるゲーム

1. キーボードから入力した4桁の数字を画面上に出力するプログラム
2. 乱数を使って、コンピュータ側の4桁の数字を作って、画面上に表示するプログラム。
3. キーボードから入力した4桁の数字とコンピュータが作った数字を比べて、等しいときに"EQUAL"と表示するプログラム
4. キーボードから入力した4桁の数字とコンピュータが作った数

字を比べて、コンピュータ側の数字の方が大きいとき等しいときに"BIGGER"と表示するプログラム

- 5. キーボードから入力した4桁の数字とコンピュータが作った数字を比べて、コンピュータ側の数字の方が小さいとき等しいときに"SMALLER"と表示するプログラム
- 5. キーボードから入力した4桁の数字とコンピュータが作った数字を比べて、等しければ"EQUAL",コンピュータ側の方が大きければ"BIGGER",コンピュータ側の方が小さければ"SMALLER"と表示するプログラム
- 6. 5.のプログラムをキーボード側の入力の数字とコンピュータ側の数字が等しくなるまで繰り返すプログラム
- 7. (おまけ)平均してもっとも少ない回数で、コンピュータの考えた数をあてるアルゴリズムを考える。

・ (その2) 桁と使われている数字による判断。(Hit&Blow)(パターン[b])

ユーザの入力した数とコンピュータの決めた数の中で"何桁が正しい数か"と"使われている数字はいくつあっているか"ヒントに数をあてる。

- 1. キーボードから入力した4桁の数字を画面に出力するプログラム
- 2. 乱数を使って、コンピュータ側の4桁の数字を作って、画面に表示するプログラム。
- 3. キーボードから入力した4桁の数を1桁づつに分けて表示するプログラム

例input -> 1234

print -> "1" "2" "3" "4"

- 4. キーボードから入力した4桁の数の1000の位の数とコンピュータが作った4桁の数の1000の位の数が等しければ"EQUAL"と、違っていれば"NOT EQUAL"と表示するプログラム

例input -> 1234

computer -> 1932

print -> EQUAL

- 5. キーボードから入力した4桁の数の100の位の数とコンピュータが作った4桁の数の100の位の数が等しければ"EQUAL"と、違っていれば"NOT EQUAL"と表示するプログラム

例input -> 1234

computer -> 1932

print -> NOT EQUAL

- 6. キーボードから入力した4桁の数の10の位の数とコンピュータが作った4桁の数の10の位の数が等しければ"EQUAL"と、違っていれば"NOT EQUAL"と表示するプログラム

例input -> 1234

computer -> 1932

print -> EQUAL

- 7. キーボードから入力した4桁の数の1の位の数とコンピュータ

が作った4桁の数の1の位の数が等しければ"EQUAL"と、違っていれば"NOT EQUAL"と表示するプログラム

例input -> 1234

computer -> 1932

print -> NOT EQUAL

- 8. キーボードから入力した4桁の数とコンピュータが作った4桁の数を比べて、桁と数が一致している数を表示するプログラム

例input -> 1234

computer -> 1932

print -> 2

- 9. キーボードから入力した4桁の数の中で0が使われていれば"USED"を、使われていなければ"UNUSED"を表示するプログラム
- 10. キーボードから入力した4桁の数の中で1が使われていれば"USED"を、使われていなければ"UNUSED"を表示するプログラム
- 11. キーボードから入力した4桁の数の中で2が使われていれば"USED"を、使われていなければ"UNUSED"を表示するプログラム
- 12. キーボードから入力した4桁の数の中で3が使われていれば"USED"を、使われていなければ"UNUSED"を表示するプログラム
- 13. キーボードから入力した4桁の数の中で4が使われていれば"USED"を、使われていなければ"UNUSED"を表示するプログラム
- 14. キーボードから入力した4桁の数の中で5が使われていれば"USED"を、使われていなければ"UNUSED"を表示するプログラム
- 15. キーボードから入力した4桁の数の中で6が使われていれば"USED"を、使われていなければ"UNUSED"を表示するプログラム
- 16. キーボードから入力した4桁の数の中で7が使われていれば"USED"を、使われていなければ"UNUSED"を表示するプログラム
- 17. キーボードから入力した4桁の数の中で8が使われていれば"USED"を、使われていなければ"UNUSED"を表示するプログラム
- 18. キーボードから入力した4桁の数の中で9が使われていれば"USED"を、使われていなければ"UNUSED"を表示するプログラム
- 19. キーボードから入力した4桁の数字とコンピュータが作った数字を比べて、"使われている数字はいくつあっているか"を表示するプログラム
- 20. 19.のプログラムをキーボードから入力した4桁の数とコンピュータが作った4桁の数を比べて、すべての桁の数が一致するまで繰り返すプログラム
- 21. (おまけ)平均してもっとも少ない回数で、コンピュータの考

えた数をあてるアルゴリズムを考える。

次の数字のすべての並び方を表示するプログラムを作ってください。

- (1) {1}
- (2) {1,2}
- (3) {1,2,3}
- (4) {1,2,3,4}
- (5) {1,2,3,4,5}
- (6) {1,2,3,4,5,6}
- (7) {1,2,3,4,5,6,7}
- (8) {1,2,3,4,5,6,7,8}
- (9) {1,2,3,4,5,6,7,8,9}

(1)の例

output -> (1)

(2)の例

output -> (1,2),(2,1)

(3)の例

output -> (1,2,3),(1,3,2),(2,1,3),(2,3,1),(3,1,2),(3,2,1)

(ヒントその1)

再帰呼び出しというテクニックを使うと、簡単に書けるかも。

(ヒントその2)

できれば、print 分を繰り返すのは、やめておいた方がいいかも。

(9)では、9! = 362880 通りあります。

受験用に年号覚えソフトを作ってください。

入力として年号を入れると、それに当て字をつけて出力するプログラムです。

例

input 538

output Go Mi Ya

input 794

output Na Ku Yo

input 1616

output I Ro I Ro

input 298

output Ni Ku Ya

出来たら、次の数列を変換してみてください。

596 213

プログラムでパズルを解いてください。

結構有名なパズル。

- 父、母、息子\*2、娘\*2、執事、犬を船を使って向こう側の岸まで渡したい。

- 船の定員は二人。(犬も一匹に数えます。)

- 船を漕げるのは、父、母、執事のみ。

- 父と母は一緒か、父と母が分かれているときは息子と娘はそれぞれの親と一緒にする。

(母がいないときに、父と娘と一緒にしない)

(父がいないときに、母と息子を一緒にしない)

- 犬以外いないか、犬と執事を一緒にする。

英語版アナグラム・ゲーム

元になる単語から、他の単語を取り出すゲームです。ゲームにはバリエーションがいろいろありますが、今回は元になる単語のすべての文字を使って、他の単語を作ることになります。

たとえば、"suer"を元の単語とすると、

suer -> rues, sure, user, ruse

と、他の単語を探すゲームです。

とりあえず、元の単語からすべての文字の組み合わせを作って、答えの候補を作るところまでを実装してみてください。

つまり、

sure -> suer, srue, sreU, seur, seru

usre, user, urse, ures, uesr, uers

rsue, rseu, ruse, rues, resu, reus

esur, esru, eusr, eurs, ersu, erus

と表示できるようなプログラムを考えてください。

(ヒントその1)

文字列を配列にしまっておけば、数字の組み合わせを作るプログラムが使えるかも。数字の組み合わせを作るプログラムで配列の引数の組み合わせは作れるから.....

(ヒントその2)

他の単語、たとえば、"seat"とか"marc"でもやってみると？

(ヒントその3)

辞書チェックまで自動化したければ、UNIXなら/usr/dict/wordsのような辞書ファイルがあります。後は、過去形や、人称変化などをチェックするかどうかです。

(ヒントその4)

余力のある人は、ぜひ単語を構成している文字から何文字かを使って、他の単語を作る問題に変えて作ってみてください。(今度は、全部の文字を使う必要がない)

例:

suer -> rues, sure, user, ruse, res, rue, sue use

marc -> car, mar, ram, mac, arc, arm, cam, cram

(ヒントその5)

今までの例がすべて4文字だからといって、4文字にこだわる必要はないです。

# 当然ですが。 :-P

プログラムをきれいに整形するソフト Minikui を作ってください。

このプログラムには、以下の機能を持たせてください。

- インデントをはずしてあげる機能
- 行詰めをしてあげる機能
- 整形後も、ちゃんとコンパイル(or実行)出来ること

このプログラムを使うことによって、きれいに箱形に整形されたプログラムのソースを作ることができます。効果として、プリントアウト時の紙資源の節約や、見掛けステップ数の削減などができます。

なお、余力のある人はオプションとして、

- ジャスティフィケーション(英文ワープロなどにある、左右を自動的にそろえるモード)
- 変数名をすべて英数1文字に変換してあげる機能

などを追加してあげると、喜ばれるでしょう。

プログラムでつるかめ算を解いてください。

「鶴と亀があわせて20匹います。足の数の合計は56本です。鶴は何匹、亀は何匹いますか。」

さらに、一般化して

「動物AとBをあわせてx匹います。AとBの足の数の合計はy本です。Aの一匹あたりの足の数はa本、Bの一匹あたりの足の数はb本です。x,y,a,bが与えられたとき、Aは何匹、Bは何匹いるか」を(解があれば)求めるプログラムを書いてください。

はじめの問題は、 $x = 20$ ,  $y = 56$ ,  $a = 2$ ,  $b = 4$ の解。