



Title	計算機プログラミングI・同演習 講義ノート2007
Author(s)	井上, 純一
Description	2007年度前期に開講された工学部情報エレクトロニクス学科2年生を対象としたLinuxシステム、C言語プログラミングに関する入門的な講義・演習の講義ノートです。この講義・演習で扱わない、より進んだ内容は後期に開講される「計算機プログラミングII」にて学習します。
Issue Date	2007-08-22T04:23:05Z
Doc URL	https://hdl.handle.net/2115/28047
Rights(URL)	https://creativecommons.org/licenses/by-nc-sa/2.1/jp/
Type	learning object
File Information	ProgI2007_exam2.pdf, 期末試験



注意事項 ~ はじめる前にお読みください ~

- 解答用紙には氏名・学籍番号を楷書で丁寧に書くこと.
- **問題 1** ~ **問題 4** の計 4 問の全てに答えること. 制限時間 90 分.
- 試験開始 30 分間は退出できない. 試験開始から 30 分経過した後の入室を認めない.
- 不明な点がある場合には黙って挙手する. 出題内容によっては答えられないものもある.
- 解答用紙の裏面を使っても良い. その場合には「裏につづく」と記すこと.

問題 1

下記のようなディレクトリ構造を持つ Linux システムに関して問い (1) ~ (4) に答えよ.

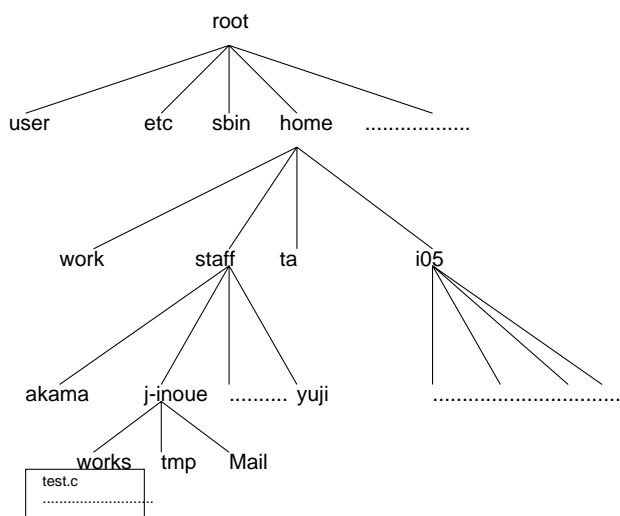


図 1: ここで問題とする Linux システムのディレクトリ構造.

- (1) j-inoue の直下にあるディレクトリ works に置かれているファイル test.c を絶対パス指定で書け.
- (2) 自分が j-inoue にいる場合, (1) のファイルを相対パス指定で書け.
- (3) 自分のホームディレクトリが j-inoue であり, カレントディレクトリはホームディレクトリの直下の works としよう. このとき, ディレクトリ akama にある sample.dat というテキストファイルを自分のカレントディレクトリにコピーするためのコマンドを書け.
- (4) (3) での操作で, このファイル sample.dat へのアクセスがシステムによって拒否された場合, どのような理由が考えられるか述べよ. また, この問題を回避するために sample.dat の所有者が行うべき操作コマンドを書け.

問題 2

下記の不完全なプログラムに関して

- (1) コメント行に記された (A) の操作を行わせる関数を書け.
- (2) プロトタイプ宣言に記された形式の引数と出力型を持つ「2つの整数のうち大きいほうの数を返す」という内容の関数本体を書け. ただし, 両者が等しい場合にはその数を返すようにすること.

```
#include<stdio.h>
#include<math.h>
int max(int a, int b); /* プロトタイプ宣言 */
main()
{
    int x,y,c;
    /* (A) x,y の値をキーボード入力する */
    c = max(x,y);
    printf("max(%d, %d) = %d\n", x,y,c);
}
```

- (3) (1)(2) で扱ったプログラムの `max(x,y)` を関数ではなく, 引数付きマクロとして書き直せ.

問題 3

- (1) 以下で $n!$ の値を再帰的関数定義で求め, その値を返す関数 `fact2` を作成し, プログラムを完成させよ.

```
#include<stdio.h>
#include<math.h>
int fact2(int n);

main()
{
    int a;
    printf("a=");
    scanf("%d",&a);
    printf("%d\n", fact2(a));
}

/* 再帰的関数定義で n! を返す関数をここに作る */
```


問題 4

以下のアルゴリズムを C 言語でコーディングせよ.

シェルピンスキー・ガスケット作成のアルゴリズム

- (1) 頂点を $(X_1, Y_1) = (0, 0)$ (点 O), $(X_2, Y_2) = (2l, 0)$ (点 A), $(X_3, Y_3) = (l, \sqrt{3}l)$ (点 B) とする正三角形を考える.
- (2) この正三角形内部の任意の点 $P^{(0)} = (x_0, y_0)$ を選ぶ.
- (3) 数 0,1,2 をランダムに選び
 - $0 \Rightarrow$ O と $P^{(0)}$ を結ぶ線分の中点に点 $P^{(1)} = (x_1, y_1)$ を置く.
 - $1 \Rightarrow$ A と $P^{(0)}$ を結ぶ線分の中点に点 $P^{(1)} = (x_1, y_1)$ を置く.
 - $2 \Rightarrow$ B と $P^{(0)}$ を結ぶ線分の中点に点 $P^{(1)} = (x_1, y_1)$ を置く.
- (4) プロセス (3) を十分多数回繰り返す.

ただし, プログラムで用いる $[0, 1]$ 擬似一様乱数は関数 $\text{RAND}()$ を用いること (自分で作らなくて良い). この関数はメイン関数の中で呼ばれる毎に異なる乱数を生成し, その値を返すものとする. また, x_0, y_0, l , 及び, 繰り返し回数などは各自が適当に決めてよい. その際, 置かれる点を `result.dat` という名前のファイルに書き込むようにすること.