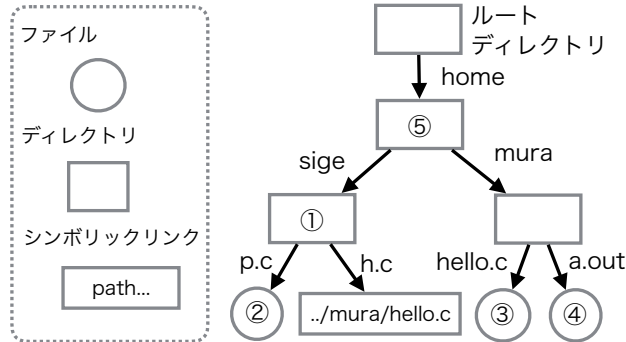


1 ファイルシステムに関する問題

次の図をよく見て答えなさい。



1. カレントディレクトリが **sige** (①) の時, ②~③を表すパスを答えなさい。(4点×3問=12点)

(a) ②のパス

相対: **p.c**

絶対: **/home/sige/p.c**

(b) ③のパス (シンボリックリンクを使用しない)

相対: **../mura/hello.c**

絶対: **/home/mura/hello.c**

(c) ③のパス (シンボリックリンクを使用して)

相対: **h.c**

絶対: **/home/sige/h.c**

2. カレントディレクトリが①のとき, ④を指すシンボリックリンク **a** をディレクトリ①に作成するコマンドを答えなさい。(3点)

\$ ln -s ../mura/a.out a

3. カレントディレクトリが⑤のとき, ④を指すシンボリックリンク **b** をディレクトリ①に作成するコマンドを答えなさい。(3点)

\$ ln -s ../mura/a.out sige/b

4. カレントディレクトリが①のとき, ④を指すハードリンク **c** をディレクトリ①に作成するコマンドを答えなさい。(3点)

\$ ln ../mura/a.out c

5. カレントディレクトリが⑤のとき, ④を指すハードリンク **d** をディレクトリ①に作成するコマンドを答えなさい。(3点)

\$ ln mura/a.out sige/d

6. 最初のカレントディレクトリが①のとき, 次の一連の操作を行った。空欄に表示されるものを答えなさい。エラーになる場合は「エラー」と答えなさい。(2点×5問=10点)

```

1 $ pwd
2 ### (a) ###
3 $ cd ../mura
4 $ pwd
5 ### (b) ###
6 $ cat hello.c
7 #include <stdio.h>
8 int main() {
9     printf("hello\n");
10    return 0;
11 }
12 $ cd ../sige
13 $ cat h.c
14 ### (c) ###
15 $ cc h.c      <-- a.out ができる
16 $ ./a.out
17 hello
18 $ chmod 600 a.out
19 $ ./a.out
20 ### (d) ###
21 $ chmod 500 a.out
22 $ ./a.out
23 ### (e) ###

```

(a) **/home/sige**

(b) **/home/mura**

(c) 同じ内容が表示されている行を答えなさい。

7行から11行

(d) **エラー**

(e) **hello**

2 低水準入出力

1. open システムコール (3 点 × 4 問 = 12 点)

別紙のプログラム p1.c を実行した結果について答えなさい。

```
1 $ cc -o p1 p1.c
2 $ ./p1
3 ### (a) ###
4 $ touch a.txt
5 $ ls -l a.txt
6 -rw-r--r-- 1 sige staff 128 ... 省略... a.txt
7 $ cat a.txt
8 $ ./p1
9 $ cat a.txt
10 ### (b) ###
11 $
```

実行結果の 4 行でファイル a.txt を作成しました。
それまでファイル a.txt はありませんでした。

(a) 実行結果 3 行に表示される内容を答えなさい。

a.txt: No such file or directory

(b) 実行結果 10 行に表示される内容を答えなさい。

abc

(c) 実行結果 3 行に何も表示されないようにするには、プログラム p1.c 7 行の open() の引数をどのように書いたら良いか答えなさい。(以下に open のカッコ内だけ書く)

**fname,
O_WRONLY|O_CREAT,
0644**

(d) 実行結果 6 行の左端の表示が -rwxr-x--- になるためには、プログラム p1.c 7 行の open() の引数をどのように書いたら良いか答えなさい。

**fname,
O_WRONLY|O_CREAT,
0750**

2. read システムコール (3 点)

次は別紙のプログラム p2.c の実行結果です。

```
$ ls -l a.dat
-rw-r--r-- 1 sige staff 128 ... a.dat
$ ./p2
### (a) ###
$
```

実行結果の空欄 (a) に何が表示されるか答えなさい。(ヒント: a.dat のサイズは 128 バイト, 表示は複数行かもしれない)

**50
50
28**

3. write システムコール (3 点 × 2 問 = 6 点)

別紙のプログラム p3.c を読んで答えなさい。

(a) このプログラムは何をするプログラムか説明しなさい。

b.txt という名前で、文字 a が 1024 個格納されたファイルを作るプログラムである。

(b) 既に b.txt という名前で大きさが 2KiB のファイルが存在した時、open システムコールの引数に O_TRUNC が無かったらどうなるか説明しなさい。

b.txt ファイルの後半 1KiB がプログラム実行前の状態のまま残る。

4. close システムコール (3 点)

別紙のプログラム p3.c で、close システムコールを書き忘れたら実行結果にどのような変化が生じるか (生じないか)。理由を付けて説明しなさい。

プロセスが終了する時、全てのオープン中ファイルは自動的にクローズされるので、close システムコールが実行されなくても実行結果に変化はない。

3 ファイル操作システムコール

以下では#include 行を省略してプログラムを書いても良い。

1. 次の連続した操作と同じことをする C 言語プログラム (p4.c) を書きなさい。作成するディレクトリの名前は A に固定で良い。但し、システムコールのエラーを表示できること。(5 点)

以下の操作と同じことをする p4 を作る

```
$ mkdir A
$ chmod 755 A
-----
# p4 の実行例
$ p4
A: File exists
$ rmdir A
$ ls -ld A
ls: A: No such file or directory
$ p4
$ ls -ld A
drwxr-xr-x 2 sige staff ... A
```

```
// p4.c
#include <stdio.h>
#include <sys/stat.h>

char *dirname = "A";
int main() {
    if (mkdir(dirname, 0755)<0) {
        perror(dirname);
        return 1;
    }
    return 0;
}
```

2. 次の操作と同じことをする C 言語プログラム (p5.c) を書きなさい。操作するディレクトリやファイルの名前は固定で良い。但し、システムコールのエラーを表示できること。(5 点)

以下の操作と同じことをする p5 を作る

```
$ mv a.txt b.txt
```

```
// p5.c
#include <stdio.h>

int main() {
    if (rename("a.txt", "b.txt")<0) {
        perror("rename");
        return 1;
    }
    return 0;
}
```

3. 次の操作と同じことをする C 言語プログラム (p6.c) を書きなさい。操作するディレクトリやファイルの名前は固定で良い。但し、システムコールのエラーを表示できること。(5 点)

以下の操作と同じことをする p6 を作る

```
$ ln -s a.txt b.txt
```

```
// p6.c
#include <stdio.h>
#include <unistd.h>

char *dst="b.txt";
int main() {
    if (symlink("a.txt", dst)<0) {
        perror(dst);
        return 1;
    }
    return 0;
}
```

4 シグナル

別紙のプログラム p7.c について答えなさい。

1. このプログラムに `hdr1` と表示させるには、どのような実行の仕方（操作）をしたら良いか答えなさい。（3点）

プログラムを起動後、5秒以内に
Ctrl-C を入力する。

2. このプログラムに `hdr2` と表示させるには、どのような実行の仕方（操作）をしたら良いか答えなさい。（3点）

プログラムを起動後、5秒以上待つ。

3. 別のターミナルにコマンドを入力してこのプログラムに `hdr1` と表示させるには、どのような手順で操作したらよいか、使用するコマンドやシグナルの名前を明記して答えなさい。（3点）

まず、`ps` コマンドを使用して `p7` を実行しているプロセスの番号を調べる。次に、`kill` コマンドを使用して `INT` シグナルを `p7` プロセスに送る。

5 環境変数

1. 入力を答えなさい。（3点 × 3問 = 9点）

```
$ printenv LANG
ja_JP.UTF-8
$ printenv LC_TIME
$ ### (1) ###
$ printenv LC_TIME
C
$ ### (2) ###
2018 年 6 月 9 日 土曜日 18 時 29 分 15 秒 JST
$ date
```

```
Sat Jun 9 18:29:38 JST 2018
$ ###(3)###
$ printenv LC_TIME
$ date
2018 年 6 月 9 日 土曜日 18 時 30 分 35 秒 JST
$
```

- (a) 空欄 (1) に何を入力したか答えなさい。

`export LC_TIME=C`

- (b) 空欄 (2) に何を入力したか答えなさい。

`env LC_TIME=ja_JP.UTF-8 date`

- (c) 空欄 (3) に何を入力したか答えなさい。

`unset LC_TIME`

2. 表示を答えなさい。（3点 × 3問 = 9点）

```
$ export X=/bin
$ X=$X:/usr/bin
$ printenv X
###(1)###
$ echo $X
###(2)###
$ export Y=$X:$X
$ printenv Y
###(3)###
$
```

- (a) 空欄 (1) に表示される内容を答えなさい。

`/bin:/usr/bin`

- (b) 空欄 (2) に表示される内容を答えなさい。

`/bin:/usr/bin`

- (c) 空欄 (3) に表示される内容を答えなさい。

`/bin:/usr/bin:/bin:/usr/bin`

別紙 (プログラムソースコード)

```
1 // p1.c
2 #include <stdio.h>
3 #include <fcntl.h>
4 #include <unistd.h>
5 int main(int argc, char*argv[]) {
6     char *fname="a.txt";
7     int fd = open(fname, O_WRONLY);
8     if (fd<0) {
9         perror(fname);
10        return 1;
11    }
12    write(fd, "abc\n", 4);
13    close(fd);
14    return 0;
15 }
```

```
1 // p2.c
2 #include <stdio.h>
3 #include <fcntl.h>
4 #include <unistd.h>
5 #define BLEN 50
6 char buf[BLEN];
7 ssize_t len; // 64bit 整数
8 int main(int argc, char*argv[]) {
9     char *fname="a.dat";
10    int fd = open(fname, O_RDONLY);
11    if (fd<0) {
12        perror(fname);
13        return 1;
14    }
15    while ((len=read(fd, buf, BLEN))>0) {
16        printf("%ld\n",len); // %ld は 64bit 整数用
17    }
18    close(fd);
19    return 0;
20 }
```

```
// p3.c
#include <stdio.h>
#include <fcntl.h>
#include <unistd.h>
int main(int argc, char*argv[]) {
    char *fname="b.txt";
    int fd = open(fname,
                  O_WRONLY|O_TRUNC|O_CREAT,
                  0644);
    if (fd<0) {
        perror(fname);
        return 1;
    }
    for (int i=0; i<1024; i++) {
        write(fd, "a", 1);
    }
    close(fd);
    return 0;
}
```

```
// p7.c
#include <stdio.h>
#include <signal.h>
#include <unistd.h>
void hdr1(int n) {
    write(2, "hdr1\n", 5);
}
void hdr2(int n) {
    write(2, "hdr2\n", 5);
}
int main() {
    signal(SIGINT, hdr1);
    signal(SIGALRM, hdr2);
    alarm(5);
    pause();
    return 0;
}
```