

1 プログラムのトレース

付録の C 言語プログラム等を実行した時の出力を書きなさい。(5 点 ×10 問=50 点)

1. \$./p0 p1

CLASS=IE4

2. \$ env CITY=SYUNAN ./p0 p1

CLASS=IE4

3. \$./p0 p2

CLASS=IE4

CAMPUS=TOKUYAMA

4. \$./p0 p3

CLASS=IE4

5. \$./p0 p4

CLASS=IE3

6. \$./p5

A

B

D

E

7. \$./p6

A

B

D

E

8. \$./p7

11,11

12,11

13,11

9. \$./p8

abc def

10. \$./p9; cat a.txt

abc def

注:「./p9; cat a.txt」のように、複数のコマンドを;で接続して一行に入力すると、1つ目がコマンドが終了し次第、次のコマンドが実行される。(コマンドの連続実行機能)

2 プログラムの改造

2.1 execvp を exeve に変更

p8.c プログラムの、8 行の `execvp` を `exeve` に変更します。正しく動作する変更後の 8 行の内容を次に書きなさい。なお、`echo` プログラムは `/bin` ディレクトリにあります。(10 点)

```
exeve("/bin/echo", a, environ);
```

2.2 正しく繰り返すように

p9.c プログラムの、13 行の `execvp` は一度しか実行されません。`echo` プログラムが 4 回実行されるように、12 行から 14 行の部分を書き換えた結果を答えなさい。(エラー処理などは省略して答えて良い) (10 点)

```
for (int i=0; i<4; i++) {  
    if (fork()==0) {  
        execvp("echo", a);  
    }  
    int stat;  
    wait(&stat);  
}
```

3 シェルプログラムの改造

付録の myshell.c に機能を追加します。新しい関数を追加しないで改造する方法を示すこと。

3.1 setenv コマンドの追加

次の実行例のように動作する、setenv コマンドを追加します。myshell.c に追加するプログラムと追加場所を答えなさい。なお、できるだけのエ

ラー処理を行うこと。(15 点)

```
$ myshell3
Command: printenv A
Command: setenv A B <--- A を作る
Command: printenv A
B
Command: setenv A C <--- A を上書きする
Command: printenv A
C
Command: setenv A B C <--- C は無視
Command: printenv A
B
```

第 31 行の前に次のプログラムを追加する。

```
} else if (strcmp(args[0], "setenv")==0) {
    if (args[1]==NULL||args[2]==NULL)
        fprintf(stderr, "setenv の引数が不足\n");
    else if (setenv(args[1], args[2], 1)!=0)
        perror("putenv");
```

3.2 env コマンドのような機能の追加

次の実行例のように動作する, env コマンドのような機能を myshell に追加します. myshell.c の何行から何行までを, どのように書き換えたら良いか答えなさい. (15 点)

```
$ ./myshell
Command: date
Mon Sep 24 21:37:15 JST 2018
Command: LC_TIME=ja_JP.UTF-8 date
2018 年 9 月 24 日 月曜日 21 時 37 分 31 秒 JST
Command: TZ=Cuba date
Mon Sep 24 08:37:42 CDT 2018
Command: LC_TIME=ja_JP.UTF-8 TZ=Cuba date
2018 年 9 月 24 日 月曜日 08 時 38 分 02 秒 CDT
Command: CLASS=IE4 printenv CLASS
IE4
Command: printenv CLASS
Command:
```

第 37 行から, 第 41 行を次のように書き換える.

```
if (pid==0) {
    int i;
    for (i=0; args[i]!=NULL; i++) {
        if (putenv(args[i])!=0) break;
    }
    if (args[i]!=NULL) {
        execvp(args[i], &args[i]);
        perror(args[i]);
    }
    exit(1);
}
```

付録

以下のプログラムは出題用の簡単版なので、エラーチェックとエラー処理を省略している場合がある。

```
// p0.c
#include <stdio.h>
#include <unistd.h>
int main(int argc, char *argv[]) {
    char *e[] = {"CLASS=IE4", NULL};
    char *a[] = {argv[1], NULL};
    execve(argv[1], a, e);
    perror(argv[1]);
    return 1;
}
```

/* p0 プログラムは、次の実行例のように
環境変数を CLASS=IE4 だけにした上で
引数で指定されたプログラムを実行する

実行例：(printenv を実行した場合)
\$ p0 /usr/bin/printenv
CLASS=IE4
\$
*/

```
// p1.c
#include <stdio.h>
extern char** environ;
int main() {
    for (int i=0; environ[i]!=NULL; i++) {
        printf("%s\n", environ[i]);
    }
    return 0;
}
```

```
// p2.c
#include <stdio.h>
#include <stdlib.h>
extern char** environ;
int main() {
    setenv("CAMPUS", "TOKUYAMA", 0);
    for (int i=0; environ[i]!=NULL; i++) {
```

```
        printf("%s\n", environ[i]);
    }
    return 0;
}
```

```
// p3.c
#include <stdio.h>
#include <stdlib.h>
extern char** environ;
int main() {
    setenv("CLASS", "IE3", 0);
    for (int i=0; environ[i]!=NULL; i++) {
        printf("%s\n", environ[i]);
    }
    return 0;
}
```

```
// p4.c
#include <stdio.h>
#include <stdlib.h>
extern char** environ;
int main() {
    setenv("CLASS", "IE3", 1);
    for (int i=0; environ[i]!=NULL; i++) {
        printf("%s\n", environ[i]);
    }
    return 0;
}
```

```
// p5.c
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/wait.h>
int main() {
    printf("A\n");
    if (fork()==0) {
        printf("B\n");
        exit(0);
        printf("C\n");
    } else {
        int stat;
        wait(&stat);
        printf("D\n");
    }
}
```

システムプログラミングⅡ H30年度 前期末試験 (2018.09.27 重村 哲至)

IE4 番 氏名 模範解答

(6/ 7)

```
printf("E\n");
return 0;
}
```

```
// p6.c
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/wait.h>
int main() {
    printf("A\n");
    if (fork()==0) {
        execl("/bin/echo", "echo", "B", NULL);
        printf("C\n");
        exit(1);
    } else {
        int stat;
        wait(&stat);
        printf("D\n");
    }
    printf("E\n");
    return 0;
}
```

```
// p7.c
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/wait.h>
int main() {
    int x = 10;
    int y = 10;
    for (int i=0; i<3; i++) {
        x++;
        if (fork()==0) {
            y++;
            printf("%d,%d\n", x, y);
            exit(0);
        } else {
            int stat;
            wait(&stat);
        }
    }
    return 0;
}
```

```
1 // p8.c
2 #include <stdio.h>
3 #include <stdlib.h>
4 #include <unistd.h>
5 extern char **environ;
6 char *a[]={"echo", "abc", "def", NULL};
7 int main() {
8     execvp("echo", a);
9     perror("echo");
10    return 1;
11 }
```

```
1 // p9.c
2 #include <stdio.h>
3 #include <stdlib.h>
4 #include <fcntl.h>
5 #include <unistd.h>
6 extern char **environ;
7 char *a[]={"echo", "abc", "def", NULL};
8 int main() {
9     close(1);
10    open("a.txt",
11         O_WRONLY|O_CREAT|O_TRUNC,0644);
12    for (int i=0; i<4; i++) {
13        execvp("echo", a);
14    }
15    return 1;
16 }
```

システムプログラミングⅡ H30年度 前期末試験 (2018.09.27 重村 哲至)

IE4 番 氏名 模範解答

(7/ 7)

```

1 // myshell.c
2 #include <stdio.h>
3 #include <stdlib.h>
4 #include <string.h>
5 #include <unistd.h>
6 #include <sys/wait.h>
7 #include <ctype.h>
8 #define MAXLINE 1000
9 #define MAXARGS 60
10
11 int parse(char *p, char *args[]) {
12     int i=0;
13     for (;;) {
14         while (isspace(*p))
15             *p++ = '\0';
16         if (*p=='\0' || i>=MAXARGS) break;
17         args[i++] = p;
18         while (*p!='\0' && !isspace(*p))
19             p++;
20     }
21     args[i] = NULL;
22     return *p=='\0';
23 }
24
25 void execute(char *args[]) {
26     if (strcmp(args[0], "cd")==0) {
27         if (args[1]==NULL)
28             fprintf(stderr, "cd の引数が不足\n");
29         else if (chdir(args[1])<0)
30             perror(args[1]);
31     } else {
32         int pid, status;
33         if ((pid = fork()) < 0) {
34             perror("fork");

```

```

35             exit(1);
36         }
37         if (pid==0) {
38             execvp(args[0], args);
39             perror(args[0]);
40             exit(1);
41         }
42         while (wait(&status) != pid)
43             ;
44     }
45 }
46
47 int main() {
48     char buf[MAXLINE+2];
49     char *args[MAXARGS+1];
50     for (;;) {
51         printf("Command: ");
52         if (fgets(buf, MAXLINE+2, stdin)==NULL) {
53             printf("\n");
54             break;
55         }
56         if (strchr(buf, '\n')==NULL) {
57             fprintf(stderr, "行が長すぎる\n");
58             return 1;
59         }
60         if (!parse(buf, args)) {
61             fprintf(stderr, "引数が多すぎる\n");
62             continue;
63         }
64         if (args[0]!=NULL) execute(args);
65     }
66     return 0;
67 }

```