

# システムプログラミング II H26 年度 前期期末試験 (2014.08.01 重村 哲至)

IE5 \_\_\_\_ 番 氏名 \_\_\_\_ 模範解答

(1/ 2)

1. 別紙 1 のプログラム (ex1.c ~ ex9.c) を読み間に答えなさい。

(1) 次の実行結果を完成しなさい。(5 点× 6 問=30 点)

```
$ ex1 aaa bbb
```

```
aaa  
bbb
```

```
$ ex3
```

```
123  
456
```

```
$ ex4
```

```
C=D  
E=F
```

```
$ ex5 aaa bbb
```

```
aaa  
bbb
```

```
$ ex7
```

```
start  
child  
parent
```

```
$ ex9
```

```
abc  
parent
```

(2) ex8 は fork 後、親子プロセスのどちらが先に実行を開始するか決まらがないので、出力に 2 種類の可能性がある。両方を書きなさい。(5 点× 2 問=10 点)

```
$ ex8
```

```
start  
fork  
fork  
child  
parent
```

```
$ ex8
```

```
start  
fork  
child  
fork  
parent
```

(3) ex6 の実行結果が次のようになるとき、ex6.c 中 /\* ※ \*/ に補うプログラムを書きなさい。(10 点)

```
$ ex6
```

```
Wed Jul 30 09:27:18 JST 2014
```

```
$ ex6 LC_TIME=ja_JP.UTF-8
```

```
2014 年 7 月 30 日 水曜日 09 時 27 分 26 秒 JST
```

```
$ ex6 LC_TIME=ja_JP.UTF-8 TZ=GMT
```

```
2014 年 7 月 30 日 水曜日 00 時 27 分 31 秒 GMT
```

ex6.c に補うプログラム

```
int i;  
for (i=1; i<argc; i++)  
    env[i-1]=argv[i];  
env[i-1]=NULL;
```

2. 別紙2のプログラム (myshell2.c) を読み問に答えなさい。

(1) 次の実行結果を完成しなさい。(5 点×4 問=20 点)

(env プログラムは、myenv と同じ動作をする。)

```
$ myshell2
Command:  echo abc
abc

Command:  pwd
/Users/sigemura/A/B
Command:  cd ..
Command:  pwd
/Users/sigemura/A

Command:  printenv LANG
ja_JP.UTF-8
Command:  env LANG=C printenv LANG
C

Command:  printenv LANG
ja_JP.UTF-8
```

(2) 実行例のような set、unset コマンドを myshell2 に追加します。

a. これらのコマンドの機能を実行するのは、親プロセスか子プロセスが答えなさい。(5 点)

親

b. myshell2 のソースプログラムで第何行の後ろに、新しいプログラムを追加すれば良いか答えなさい。(5 点)

第 25 行

実行例

```
Command: printenv A
Command: set A B
Command: printenv A
B
Command: unset A
Command: printenv A
Command:
```

c. 追加するプログラムを set コマンド、unset コマンド部分に分けて書きなさい。(10 点×2 問=20 点)

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

```
} else if (strcmp(args[0], "unset")==0) {
    if (args[1]==NULL)
        fprintf(stderr, "unset 引数が不足\n");
    else if (unsetenv(args[1])<0)
        perror(args[1]);
```

```

/* ex1.c */
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char *argv[]) {
    int i;
    for (i=1; i<argc; i++)
        printf("%s\n", argv[i]);
    exit(0);
}

-----

/* ex2.c */
#include <stdio.h>
#include <stdlib.h>
extern char **environ;
int main(int argc, char *argv[]) {
    int i;
    for (i=0; environ[i]!=NULL; i++)
        printf("%s\n", environ[i]);
    exit(0);
}

-----

/* ex3.c */
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
int main(int argc, char *argv[]) {
    execl("ex1", "ex1", "123", "456", NULL);
    exit(1);
}

-----

/* ex4.c */
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
char *arg[] = {"ex2", "A=B", NULL};
char *env[] = {"C=D", "E=F", NULL};
int main(int argc, char *argv[]) {
    execve("ex2", arg, env);
    exit(1);
}

-----

/* ex5.c */
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
char *arg[1000];
int main(int argc, char *argv[]) {
    int i;
    arg[0]="ex1";
    for (i=1; i<argc; i++)
        arg[i]=argv[i];
    arg[i]=NULL;
    execv("ex1", arg);
    exit(1);
}

-----

/* ex6.c */
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
char *env[1000], *arg[] = {"date", NULL};
int main(int argc, char *argv[]) {
    /* ※ */
    execve("/bin/date", arg, env);
    exit(1);
}

```

```

/* ex7.c */
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
int main(int argc, char *argv[]) {
    int pid;
    printf("start\n");
    pid = fork();
    if (pid==0) {
        printf("child\n");
        exit(123);
    } else {
        int stat;
        while(wait(&stat)!=pid)
            ;
        printf("parent\n");
    }
    exit(0);
}

-----

/* ex8.c */
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
int main(int argc, char *argv[]) {
    int pid;
    printf("start\n");
    pid = fork();
    printf("fork\n");
    if (pid==0) {
        printf("child\n");
        exit(123);
    } else {
        int stat;
        while(wait(&stat)!=pid)
            ;
        printf("parent\n");
    }
    exit(0);
}

-----

/* ex9.c */
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
int main(int argc, char *argv[]) {
    int pid;
    pid = fork();
    if (pid==0) {
        execl("ex1", "ex1", "abc", NULL);
        printf("child1\n");
        exit(123);
        printf("child2\n");
    } else {
        int stat;
        while(wait(&stat)!=pid)
            ;
        printf("parent\n");
    }
    exit(0);
}

```