

1 文字コードに関する問題

文字列"8月2日\n"を以下の方式で符号化した結果を16進数で答えなさい。なお、各文字に割り振られたコードは以下の表の通りである。また、Shift_JIS方式で符号化した結果を例として示す。

(5点×4問=20点)

文字	ASCII	JIS 0208	Unicode
8	0x38	-	U+0038
月	-	0x376e	U+6708
2	0x32	-	U+0032
日	-	0x467c	U+65e5
\n	0x0a	-	U+000a

Shift_JIS に符号化した例 (解答例)

0x38 0x8c 0x83 0x32 0x93 0xfa 0x0a

1. ISO-2022-JP に符号化した場合

0x38
0x1b 0x24 0x42 0x37 0x6e
0x1b 0x28 0x42 0x32
0x1b 0x24 0x42 0x46 0x7c
0x1b 0x28 0x42 0x0a

2. EUC-JP に符号化した場合

0x38 0xb7 0xee 0x32
0xc6 0xfc 0x0a

3. UTF-32BE に符号化した場合

0x00 0x00 0x00 0x38
0x00 0x00 0x67 0x08
0x00 0x00 0x00 0x32
0x00 0x00 0x65 0xe5
0x00 0x00 0x00 0x0a

4. UTF-8 に符号化した場合

0x38 0xe6 0x9c 0x88 0x32
0xe6 0x97 0xa5 0x0a

2 環境変数に関する問題

1. プログラム (p1) が実行例のように動作するようにソース・プログラム (p1.c) 中の空欄 (A) に補うプログラムを答えなさい。

(5点×2問=10点)

リスト 1: p1 のソースプログラム

```
// p1.c
#include <stdio.h>
#include <stdlib.h>
int main() {
    char *val;
    val = (A);

    if (B)
        printf("hello\n");
    else
        printf("%s\n", val);
    return 0;
}
```

リスト 2: p1 の実行例

```
$ p1
hello
$ env GOAISATU=konichiwa p1
konichiwa
$
```

(A) `getenv("GOAISATU")`

(B) `(val==NULL)`

2. 次の実行結果は前出の p1 を用いたものです。空欄 (B) に入力されたコマンドを答えなさい。(5点)

リスト 3: 実行結果

```
$ p1
konichiwa
$ (B)
$ p1
hello
$
```

`unset GOAISATU`

`GOAISATU=hello` (別解)

`export GOAISATU=hello` (別解)

3 プログラム実行に関する問題

プログラム (p2) は実行例に示すように、コマンド行引数で指定した環境変数の書換えをした上で date プログラムを実行します。環境変数の書換えはいくつでも指定できます。ソースプログラム (p2.c) 中の空欄 (C)、(D)、(E) に適切なプログラムを答えなさい。

リスト 4: p2 のソースプログラム

```
// p2.c
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
extern char **environ;
char *execArgs[] = (D);
char *execPath = "/bin/date";

int main(int argc, char *argv[]) {
    (C)
    execve(execPath, execArgs, (E));
    perror(execPath);
    return 1;
}
```

リスト 5: p2 の実行例

```
$ p2
Sun Jul 31 23:32:24 JST 2016
$ p2 LC_TIME=ja_JP.UTF-8
2016 年 7 月 31 日 日曜日 23 時 32 分 39 秒 JST
$ p2 LC_TIME=ja_JP.UTF-8 TZ=Cuba
2016 年 7 月 31 日 日曜日 10 時 32 分 52 秒 CDT
$ p2 LC_TIME=ja_JP.UTF-8 TZ=Cuba LC_TIME=ru_RU.UTF-8
в о с к р е с е н ь е , 31 и ю л я 2016 г . 10:34:50 (CDT)
$
```

1. (C) のプログラム (10 点)

```
for (int i=1; i<argc; i++)
    putenv(argv[i]);
```

2. (D) のプログラム (5 点)

```
{"date", NULL}
```

3. (E) のプログラム (5 点)

```
environ
```

4 fork-exec に関する問題

1. p3 の出力を答えなさい。(5 点)

リスト 6: p3 のソースプログラム

```
#include <stdio.h>
#include <unistd.h>
#include <sys/wait.h>
int main(int argc, char *argv[]) {
    int pid = fork();
    if (pid!=0) {
        int stat;
        wait(&stat);
        printf("親プロセス\n");
    } else {
        printf("子プロセス\n");
    }
}
```

子プロセス
親プロセス

2. p6 の出力を答えなさい。(5 点)

リスト 7: p6 のソースプログラム

```
#include <stdio.h>
#include <unistd.h>
#include <sys/wait.h>
int main(int argc, char *argv[]) {
    int pid = fork();
    if (pid==0) {
        printf("子プロセス\n");
    } else {
        int stat;
        wait(&stat);
        printf("親プロセス\n");
    }
}
```

子プロセス
親プロセス

3. p4 の出力を答えなさい。(5 点)

リスト 8: p4 のソースプログラム

```
#include <stdio.h>
#include <unistd.h>
char **environ;
```

```
char *args[]={ "echo", "aaa", "bbb", NULL };
int main(int argc, char *argv[]) {
    execve("/bin/echo", args, environ);
}
```

aaa bbb

4. p5 の出力を答えなさい。(5 点)

リスト 9: p5 のソースプログラム

```
#include <stdio.h>
#include <unistd.h>
char **environ;
char *args[]={ "echo", "aaa", "bbb", NULL };
int main(int argc, char *argv[]) {
    execve("/bin/echo", args, environ);
    printf("終了\n");
}
```

aaa bbb

5. p7 の出力を答えなさい。(5 点)

リスト 10: p7 のソースプログラム

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/wait.h>
int main(int argc, char *argv[]) {
    int x = 1;
    for (int i=0; i<3; i++) {
        int pid = fork();
        if (pid==0) {
            x=x+1;
            printf("%d\n", x);
            exit(0);
        } else {
            int stat;
            wait(&stat);
        }
    }
}
```

2
2
2

5 世界時計

右の実行例のように世界各地の現在時刻を表示するプログラム (p8) を作ります。下のプログラムは作成途中の p8 のソースコードです。tz 配列はタイムゾーンを表す文字列を格納しています。print 関数を完成しなさい。(20 点)

```
$ p8
日本:Mon Aug  1 14:09:24 JST 2016
ロンドン:Mon Aug  1 06:09:24 BST 2016
ニューヨーク:Mon Aug  1 01:09:24 EDT 2016
香港:Mon Aug  1 13:09:24 HKT 2016
$
```

```
// #include<... 省略
extern char **environ;
#define N 4
char *tz[N]={"Japan", "Europe/London", "America/New_York", "Asia/Hong_Kong" };
char *name[N] = {"日本", "ロンドン", "ニューヨーク", "香港"};
char *execPath = "/bin/date";
char *args[] = {"date", NULL};
void print(int i) {

    int pid = fork();                // fork 5 点
    if (pid<0) {
        perror("fork");             // エラー処理 1 点
        exit(1);
    } else if (pid>0) {
        int status;
        wait(&status);              // wait 4 点
    } else {
        printf("%s:", name[i]);      // printf 2 点
        fflush(stdout);              // exec 前にフラッシュ必要
        setenv("TZ", tz[i], 1);      // setenv 3 点
        execve(execPath, args, environ); // execve 3 点
        perror(execPath);            // エラー処理 2 点
        exit(1);
    }
}

int main() {
    for (int i=0; i<N; i++)
        print(i);
}
```