

1 リダイレクト (20)

標準入力 (stdin) から読み込んだデータを、(a) 英小文字を大文字に変換する、(b)rot13 で暗号化し、標準出力 (stdout) に出力するプログラムをそれぞれ作成した。

rot13 とは、ABC を NOP,XYZ を KLM といったように、A-Z の文字を 13 文字ずらす、英小文字も abc→nop,xyz→klm に変換する、原始的な暗号化。

<pre>// (a) uc.c -> uc.exe void main() { int c ; while((c = getchar())) { if () { c = c - 'a' + 'A' ; } putchar(c) ; } } ----- // file.txt Hello \$ uc.exe < file.txt ~~~~~(E)</pre>	<pre>// (b) rot13.c -> rot13.exe void main() { char buff[1000] ; while(fgets(buff, sizeof(buff), stdin) != NULL) { char *pc ; for() { if (*pc >= 'A' && *pc <= 'Z') *pc = (*pc - 'A' + 13) % 26 + 'A' ; else if (*pc >= 'a' && *pc <= 'z') *pc = } } fputs(buff , stdout) ; } } [設問 1] 下線部 (A) ~ (D) に相応しい処理を 答えよ。(3x4) [設問 2] 左に示すように実行した場合、 結果 (E),(F) を答えよ。 ^Z は、標準入力の終了とする。(4x2)</pre>
<pre> \$ rot13.exe Yes ^Z ~~~~~(F)</pre>	

2 説明問題 (20)

- 前設問のプログラムを使い、以下の様にプログラムを実行した。(A),(B) の結果を答えよ。(5x2)

```
$ rot13.exe < file.txt >tmp.out || $ rot13.exe < file.txt | rot13.exe
$ uc.exe < tmp.out ||
||
||
||
~~~~~(A) || ~~~~~(B)
```

- 出力リダイレクトと、パイプ機能について説明せよ。(5x2)

3 構造体の理解 (20)

学生 Student が、科目 Subject で、何点だったか Result を、以下の様に表現する。

学生 Student の要素は、学籍番号 uid, 名前 name, 受講年 year を表す。科目 Subject は、科目番号 sid, 科目名 subject, 担当教員名 teacher。結果 Result は、学籍番号 uid(Student の uid と対応)、科目番号 sid(Subject の sid と対応)、点数 point を意味する。

```

struct Student {          | struct Student stu[ 3 ] = {
    int  uid ;            |   { 1001 , "tomoko" , 2003 } ,
    char name[ 20 ] ;    |   { 1002 , "mitsuki" , 2014 } ,
    int  year ;          |   { 1003 , "ayuka" , 2016 } ,
};                          | };
struct Subject {         | struct Subject sub[ 3 ] = {
    int  sid ;           |   { 2001 , "Programming" , "saitoh"  } ,
    char subject[ 20 ] ; |   { 2002 , "Mathematics" , "nakatani" } ,
    char teacher[ 20 ] ; |   { 2003 , "Science" , "katoh"  } ,
};                          | };
struct Result {          | struct Result result[ 3 ] = {
    int  uid ;           |   { 1001 , 2002 , 80 } , // tomoko 2013 Mathematics nakatani 80
    int  sid ;           |   { 1003 , 2001 , 75 } , // ayuka 2016 Programming saito 75
    int  point ;        |   { 1002 , 2003 , 92 } , // mitsuki 2014 Science katoh 92
};                          | };

```

```

void test() {
    struct Subject* sj ;
    printf( "%d\n" , stu[2].year ) ;
    printf( "%s\n" , sub[1].teacher ) ;
    printf( "%d\n" , result[0].point ) ;
    printf( "%d\n" , (*(stu+1)).uid ) ;
    sj = sub + 2 ;
    printf( "%c\n" , sj->subject[1] ) ;
    sj-- ;
    printf( "%s\n" , (*(sj-1))->teacher ) ;
    printf( "%d\n" , stu->uid ) ;
}

```

[設問] 関数 test() を実行した時
表示される結果を答えよ。
エラーが発生する場合は×を記載

----- (2)
----- (3)
----- (3)
----- (3)
----- (3)
----- (3)
----- (3)

4 プログラム記述 (20)

前設問のデータ構造で、配列 struct Result result[] の内容を、データ右のコメント欄の様に出力したい。そのプログラムを記述せよ。

5 オブジェクト指向と用語 (20)

<pre> class Pokemon { private: char name[10] ; char type ; int hp ; public: Pokemon(const char s[] , char t , int h) { strcpy(name , s) ; type = t ; hp = h ; } void print() { printf("%s %c %d\n" , name , type , hp) ; } }; void main() { Pokemon pika("Pikachu" , 'E' , 35) ; pika.print() ; } </pre>	<p> [設問 1] プログラムの実行結果を答えよ。(4)</p> <p> [設問 2]</p> <p> 左に示すようなオブジェクト指向による</p> <p> プログラムの説明文の下線部に相応しい</p> <p> 用語を答えよ。(4x4)</p> <p> オブジェクト指向では、(A)_____と</p> <p> (B)_____を一つのクラス内に併記する。</p> <p> print() のような関数は (C)_____</p> <p> と呼ばれ、特にデータ初期化の関数は、</p> <p> (D)_____ と呼ぶ。"private"以下に</p> <p> 書いてある要素は、(C) 以外の場所で使う</p> <p> と (F)_____ となる。</p>
---	---

6 プログラム作成 (20)

フィギュアスケートの採点は、複数の審判員のポイントの合計をする際に、最高点,最低点がそれぞれ1つ除外して計算する。

以下のように、スケータ (最大 10 名) と 5 人の審判員のデータが 1 行 1 件で与えられる。

これらのデータを全員分を格納するための構造体を宣言し、データを標準入力から読み取り、各スケータの名前とポイントを表示するプログラムを作成せよ。

```

YuzuruHanyu      8.7 7.7 8.5 8.2 9.0   Point=8.7+8.5+8.2
YuriKatsuki     7.4 8.2 8.2 8.0 6.7   Point=7.4+8.2+8.0
VictorNikiforv  8.5 8.2 8.8 8.8 8.6   Point=8.5+8.8+8.6
~Z
EOF

```