

2015 年度プログラミング入門 中間試験問題 ※ 全ての解答は解答用紙に記入のこと。

問題 1 ((1)～(6)各 2 点, (7)～(12)各 1 点, 計 18 点)

(1)～(3)の空欄(下線部)に当てはまる語を選びなさい。

(1)演習中、標準で使用している C コンパイラは_____である。

{ア: emacs, イ: gcc, ウ: gdb, エ: vi }

(2)コンピュータの基本ソフトウェアのことを_____と呼ぶ。

{ア: CPU, イ: DRAM, ウ: iMac, エ: OS }

(3)C 言語は_____に区分される。

{ア: 自然言語, イ: 機械語, ウ: 高級言語, エ: インタプリタ言語}

次の(4)～(6)文中の空欄(下線部)を埋めなさい。

(4)倍精度浮動小数点型の数 x を標準入力し、その x を小数点以下切り捨てで標準出力する 2 行の文。

```
scanf( 4-a );
```

```
printf( 4-b );
```

(5) 倍精度浮動小数点型で要素数 5 の配列 student[] を宣言し、各要素を 1.0 で初期化する文。

```
double student[5] = 5;
```

(6) 宣言かつ初期化されている倍精度浮動小数点型の数 x について、小数点以下の部分を表示する文。

```
printf(“%f\n”, x - 6);
```

次の(7)～(12)宣言文で文法上正しいものに○を、誤っているものには×を、それぞれ答えなさい。

(7) double _aaaa;

(8) int aa0;

(9) int a b c;

(10) double int[100];

(11) double i;

(12) int a\$\$a;

問題 2 (各 2 点 × 5 = 計 10 点)

```
int i = -3145
```

```
double a = 3.1415
```

であるとき、printf による標準出力が例の通りになるよう、次の書式指定の空欄[]を埋めなさい。

```
printf(“(1):[ ]\n”, i);
```

```
printf(“(2):[ ]\n”, i);
```

```
printf(“(3):[ ]\n”, a);
```

```
printf(“(4):[ ]\n”, a);
```

```
printf(“(5):[ ]\n”, a);
```

ただし、小数点以下の標準表示桁数は 6 とする。また、表示桁数の制御は全て printf の書式指定で行うこと。空白文字を直接表示させるような方法は正答としない。

出力例 (空白は_で示す。__は空白 3 つ分である) :

(1):[__-3145]

(2):[-3145__]

(3):[__3.142]

(4):[3.141500E+00]

(5):[+3.141500]

問題 3 ((1)各 2 点 × 4, (2)各 1 点 × 6, 計 14 点)

(1) 以下のプログラムの実行結果はどうなるか？

なお、%f の表示桁数は小数点以下 2 桁だとする。

b = ____ (ア) ____

m = ____ (イ) ____

c = ____ (ウ) ____

n = ____ (エ) ____

```
#include <stdio.h>
```

```
int main() {
```

```
    int k = 5, m, n;
```

```
    double a = 3.14, b, c;
```

```
    b = k + a; /* (あ) */
```

```
    m = k + a;
```

```
    c = m;
```

```
    n = b;
```

```
    printf(“b = %f\n”, b);
```

```
    printf(“m = %d\n”, m);
```

```
    printf(“c = %f\n”, c);
```

```
    printf(“n = %d\n”, n);
```

```
    return 0;
```

```
}
```

(2) 以下の文章の空欄(A)～(G)に適する語句又は式を、
下の語群(ア)～(サ)から選び記号で答えなさい
(同じアルファベットの空欄には同じ語句が入る)。

上記(1)のプログラム中のコメント/*(あ)*/が記されている行は __ (A) __ 演算による __ (B) __ の __ (C) __ が行われている。この行を __ (D) __ __ (C) __ を使用して書くと $b =$ __ (E) __ となる。なお、この __ (D) __ __ (C) __ の事を __ (F) __ と呼ぶ。

ア. 四則, イ. 明白, ウ. $\text{double}(k) + a$, エ. 暗黙,
オ. キャスト, カ. 明示的, キ. 代入, ク. 混合,
ケ. キャスター, コ. $(\text{double})k + a$, サ. 型変換

問題 4 (各 2 点 × 7 = 14 点)

以下は、読み込んだ 2 つの整数値が両方とも奇数か、片方が偶数か、両方とも偶数かを判断・表示するプログラムである。下線 (1)～(4)、および、複数行からなる空欄(5)～(7)を適切に埋めなさい。

実行例：(以下、入力は斜体)

% ./a.out

2 つの整数を入力してください

整数 a : 100

整数 b : 53

片方が偶数です

%

```
#include <stdio.h>
```

```
int main() {
```

```
    int ____ (1) ____, a, b ;
```

```
    printf("2 つの整数を入力して下さい\n");
```

```
    printf("整数 a : ");
```

```
    scanf("%d", &a);
```

```
    printf("整数 b : ");
```

```
    scanf("%d", &b);
```

```
    if( __ (2) __ == 0 ) count++ ;
```

```
    if( __ (3) __ == 0 ) count++ ;
```

```
    ____ (4) ____ (count) {
```

```
        __ (5) __
```

```
        /*複数行入れる*/
```

```
        __ (6) __
```

```
        /*複数行入れる*/
```

```
        __ (7) __
```

```
        /*複数行入れる*/
```

```
    }
```

```
    return 0;
```

```
}
```

問題 5 (各 2 点 × 7 = 14 点)

下記のプログラムは、入力されたデータを逆順で表示するプログラムだが、その際ループにおいて、インクリメントとデクリメントを利用する二通りの方法により実現されている。

括弧内を埋めて、プログラムを完成させなさい。

ただし、データの最大個数は 20 個とし、データの個数は変数 n に、データは配列 data にそれぞれキーボードから入力するものとする。

実行例：(以下、入力は斜体)

% ./a.out

データの個数を入れて下さい : 6

1 2 3 4 5 6

6 5 4 3 2 1

6 5 4 3 2 1

%

```
#include <stdio.h>
```

```
int main() {
```

```
    int i, n;
```

```
    int data[20];
```

```

printf("データの個数を入れて下さい：");
scanf(__(1)__);

for (__(2)__ i++) {
    scanf(__(3)__);
}

for (__(4)__ i++) {
    printf("%d ",data[__(5)__]);
}
printf("\n");

for (__(6)__ i--) {
    printf("%d ",data[__(7)__]);
}
printf("\n");

return 0;
}

```

問題 6 (各 2 点 × 7 = 14 点)

正の整数を素数の掛け算の形で表すことを素因数分解といい、例えば 24 は 2 の 3 乗 × 3 と表せる。2 以上 50 未満の整数を素因数分解するプログラムを考えた。まず 2 で割り算を繰返して割り切れる回数を数え、割れなくなったらさらに 3, 5, 7 で割って割り切れる回数を数えていくと、最後に残るのは 1 か素数となるので、結果を

「(2 の ○ 乗) x (3 の ○ 乗) x (5 の ○ 乗) x (7 の ○ 乗) x 残り」のように表示して終了する。上記の動作をするように、コメントに従いプログラムの空欄を埋めなさい。

実行例：(以下、入力は斜体)

% ./a.out

2 以上 50 未満の整数を入力してください: 24

(2 の 3 乗) x (3 の 1 乗) x (5 の 0 乗) x (7 の 0 乗) x 1

% ./a.out

2 以上 50 未満の整数を入力してください: 39

(2 の 0 乗) x (3 の 1 乗) x (5 の 0 乗) x (7 の 0 乗) x 13

%

プログラム：

```
#include <stdio.h>
```

```

int main() {
    int i, n;
    int m[4]=__(1)__; /* 割る数 2,3,5,7 を配列に格納 */
    int p[4]; /*それぞれ 2,3,5,7 で割りきれた回数のカウンタ */

    printf("2 以上 50 未満の整数を入力してください:");
    scanf("%d", &n); /* エラーチェックはしない */

    for ( i=0 ; i<4 ; i++ ){
        __(2)__; /* カウンタを初期化 */

        __(3)__ { /* 無限ループ */

            /* 以下の行で m[i] で割り切れるか試す */
            if ( __(4)__ ){

                __(5)__; /* 割り切れた回数をカウント */
                n = __(6)__; /* 割り算の結果 */
            }
            else break; /*割り切れなくなったらその数での
                           割り算は終わり*/
        }
    }

    for ( i=0 ; i<4 ; i++ ){
        printf("(%d の %d 乗) x ", __(7)__);
    }
    printf("%d\n", n ); /* 割り算した残り */

    return 0;
}

```

問題 7 (各 2 点 × 8 = 16 点)

以下は1から100までの数を順に表示するプログラムである。

ただし、3の倍数のときは数の代わりに「Fizz」、5の倍数のときは「Buzz」、3と5両方の倍数のときは「FizzBuzz」と表示する。

右図のフローチャートをもとに、(1)～(8)に当てはまる内容を解答欄に記入しなさい。

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    int i;
```

```
    for(i = 1; i <= 100; i++) {
```

```
        if(__(1)__)
```

```
            printf("__ (2)__ \n");
```

```
        else if(__ (3)__)
```

```
            printf("__ (4)__ \n");
```

```
        else if(__ (5)__)
```

```
            printf("__ (6)__ \n");
```

```
        else
```

```
            printf("__ (7)__ \n", __ (8)__);
```

```
    }
```

```
    return 0;
```

```
}
```

