

プログラミング II

期末テスト（ペーパーテスト）問題用紙

（'09年7月29日）

解答上、その他の注意事項

- I. 問題は、問 I～III までである。
- II. 持ち込みは不可である。筆記用具・時計・学生証以外のものは、かばんの中などにしまうこと。
- III. 解答用紙の右上の欄に学籍番号・名前を記入すること。
- IV. 解答中の文字（特に a と d）がはっきりと区別できるよう注意すること。
- V. ペーパーテストの配点は 30 点とする。（オンラインテストは 10 点）

- I. 以下のプログラムで、関数 `foo` は文字列 `str` を引数に取り、文字列の各文字のあとに 1 字ずつ空白文字を出力する。プログラム全体では、

```
h e l l o  
e v e r y o n e !
```

と出力する。(i) ~ (iii) の空欄を埋めて、プログラムを完成させよ。

```
#include <stdio.h>

void foo(char *str) {
    for ( ; (i) ; (ii) ) {
        putchar((iii));
        putchar(' ');
    }
}

int main(void) {
    foo("hello");
    putchar('\n');
    foo("everyone!");
    putchar('\n');

    return 0;
}
```

- II. 以下のプログラムで、関数 `bar` は 2 つの `int` 型のメンバ `x, y` を持つ構造体 `struct point` へのポインタを受け取り、このポインタが指し示す構造体のメンバの `x` と `y` の値を入れ替える。プログラム全体では、(2, 1) と出力する。(i) ~ (iii) の空欄を埋めて、プログラムを完成させよ。((iii) は演算子)

```
#include <stdio.h>

struct point {
    int x, y;
};

(i) bar((ii) ptr) {
    int tmp = ptr (iii) x;
    ptr (iii) x = ptr (iii) y;
    ptr (iii) y = tmp;
}

int main(void) {
    struct point pt;
    pt.x = 1; pt.y = 2;
    bar(&pt);
    printf("(%d, %d)\n", pt.x, pt.y);

    return 0;
}
```

III. 以下の多肢選択問題に答えよ。

(i) 次のプログラムの出力結果を下の選択肢より1つ選べ。

```
#include <stdio.h>

int main(void) {
    int i, j;

    for (i = 4; i >= 0; i--) {
        for (j = 0; j < i; j++) {
            printf("%d", j);
        }
        printf("¥n");
    }

    return 0;
}
```

- | | | | |
|-----------------------------|-----------------------------|-----------------------------|-----------------------------|
| (A). 3
23
123
0123 | (B). 0
11
222
3333 | (C). 0123
012
01
0 | (D). 0123
123
23
3 |
| (E). 0000
111
22
3 | (F). 3210
210
10
0 | (G). 3
32
321
3210 | (H). 0
01
012
0123 |

(ii) 次のプログラムの出力結果を下の選択肢より1つ選べ。

```
#include <stdio.h>

int main(void) {
    int i=1;

    printf("%d ", i++);
    printf("%d ", ++i);
    printf("%d ", i = i+2);
    printf("%d¥n", i);

    return 0;
}
```

- | | | | |
|--------------|--------------|--------------|--------------|
| (A). 2 2 3 5 | (B). 2 2 5 5 | (C). 1 3 3 5 | (D). 1 3 5 5 |
| (E). 1 2 3 5 | (F). 2 3 5 5 | (G). 2 3 3 5 | (H). 1 2 5 5 |

(iii) 次のプログラムの出力結果を下の選択肢より1つ選べ。

```
#include <stdio.h>

int x = 1;

void funyu(void) {
    x++;
    printf("%d ", x);
}

int main(void) {
    int i, x=5;

    x++;
    printf("%d ", x);
    for (i=0; i<2; i++) {
        int x = 3*i;
        funyu();
        x++;
        printf("%d ", x);
    }
    x++;
    printf("%d ", x);
    return 0;
}
```

- (A). 6 1 2 1 2 3 (B). 6 1 2 3 4 5 (C). 6 2 1 3 4 7
(D). 6 2 1 3 2 7 (E). 6 2 2 3 5 6 (F). 6 2 3 4 5 6

(iv) 次のプログラムの出力結果を下の選択肢より1つ選べ。

```
#include <stdio.h>

char *becho(char *str) {
    char *p = str;

    for ( ; *str; str++) {
        p = str;
    }
    return p;
}

int main(void) {
    printf(becho("Hello"));
    printf(", ");
    printf(becho("World"));
    printf("¥n");
    return 0;
}
```

- (A). hello, world (B). HELLO, WORLD (C). dlroW ,olleH
(D). ello, orld (E). o, d (F). olleH, dlroW

(v) 次のプログラムの出力結果を下の選択肢より1つ選べ。

```
#include <stdio.h>

void puhi(char *str1, char *str2) {
    while (*str1 && *str2) {
        putchar(*str1);
        str1++; str2++;
        if (*str1=='\0' || *str2=='\0') {
            break;
        }
        putchar(*str2);
        str1++; str2++;
    }
}

int main(void) {
    puhi("abcd", "wxyz");
    putchar(' ');
    puhi("ijkl", "pqr");
    putchar(' ');
    puhi("fgh", "stuv");
    putchar('¥n');
    return 0;
}
```

- (A). xbyd pjrl sgu (B). aycz iqk fthv (C). xbyd pjrl sguv
 (D). aycz iqkl fthv (E). xbyd pjr sgu (F). aycz iqk fth

(vi) 関数 henyo は、引数として与えられたポインタの指す変数を-1倍する関数である。したがって、以下のプログラムは `m=-20` と出力する。

空欄 `あ`、`い` に当てはまる組合せを以下の選択肢から1つ選べ。

```
#include <stdio.h>

void henyo(あ) {
    *p *= (-1);
}

int main(void) {
    int m = 20;
    henyo(い);
    printf("m=%d¥n", m);
    return 0;
}
```

- (A). `あ` `int p`、`い` `*m` (B). `あ` `int p`、`い` `&m`
 (C). `あ` `int *p`、`い` `*m` (D). `あ` `int *p`、`い` `&m`
 (E). `あ` `int &p`、`い` `*m` (F). `あ` `int &p`、`い` `&m`
 (G). `あ` `int &p`、`い` `m` (H). `あ` `int *p`、`い` `m`

(vii) 次のプログラムの出力結果を下の選択肢から 1 つ選べ。

```
#include <stdio.h>

void pecho(int *p, int q) {
    *p *= 3;
    q += 5;
}

int main(void) {
    int x = 1, y = 0;

    pecho(&x, y);
    printf("x=%d, y=%d\n", x, y);

    return 0;
}
```

- (A). x=1, y=5 (B). x=3, y=0 (C). x=3, y=5
(D). x=1, y=0 (E). x=5, y=3 (F). x=5, y=0

(viii) 次のプログラムの出力結果を下の選択肢から 1 つ選べ。

```
#include <stdio.h>

void bote(int *ptr) {
    *ptr += 2;
    ptr += 3;
    *ptr += 3;
    ptr += 2;
}

int data[6] = { 1, 3, 5, 7, 9, 11 };

int main(void) {
    int i;

    bote(data);
    for (i=0; i<6; i++) {
        printf("%d ", data[i]);
    }
    printf("\n");

    return 0;
}
```

- (A). 1 3 8 7 9 13 (B). 3 6 8 10 9 13 (C). 4 3 6 7 9 11
(D). 1 3 5 9 9 14 (E). 3 3 5 7 11 11 (F). 3 3 5 10 9 11

(ix) 次のプログラムの出力結果を下の選択肢から 1 つ選べ。

```
#include <stdio.h>

void swap(int *p1, int * p2) {
    int t = *p1;
    *p1 = *p2;
    *p2 = t;
}

int *hogya(int *p) {
    int i = *p;
    while(*p<=i) {
        p++;
    }

    return p;
}

int data[8] = { 6, 4, 3, 2, 5, 8, 1, 10 };

int main(void) {
    int k;
    int *ptr1 = data;
    int *ptr2;

    for (k=0; k<3; k++) {
        ptr2 = hogya(ptr1);
        swap(ptr1, ptr2);
        ptr1++;
    }

    for (k=0; k<8; k++) {
        printf("%d ", data[k]);
    }
    printf("¥n");

    return 0;
}
```

(A). 4 3 2 6 5 8 1 10

(C). 1 2 3 4 5 6 8 10

(E). 6 4 5 8 3 2 1 10

(G). 8 5 4 2 3 6 1 10

(B). 6 4 3 2 5 8 1 10

(D). 1 2 3 6 4 5 8 10

(F). 4 3 2 1 6 5 8 10

(H). 8 6 5 4 2 3 1 10

(x) 次のプログラムの空欄 ~ に当てはまる組合せを以下の選択肢から 1 つ選べ。

```
#include <stdio.h>

char *choba(char *str) {
    char *p = ;

    while (*str != ) {
        if (*str=='a') {
            p = str;
        }
        str++;
    }
    return p;
}

int main(void) {
    char *test = "abcdeabcabcxyz";
    char *p = choba(test);

    if (p != ) {
        printf(p);
    }
    return 0;
}
```

- | | | | | | | | | | | | | | |
|------|---|-----------------------------------|---|-----------------------------------|---|-----------------------------------|------|---|-----------------------------------|---|-----------------------------------|---|-----------------------------------|
| (A). | あ | <input type="text" value="NULL"/> | い | <input type="text" value="'\0'"/> | う | <input type="text" value="NULL"/> | (B). | あ | <input type="text" value="NULL"/> | い | <input type="text" value="'\0'"/> | う | <input type="text" value="'\0'"/> |
| (C). | あ | <input type="text" value="NULL"/> | い | <input type="text" value="NULL"/> | う | <input type="text" value="NULL"/> | (D). | あ | <input type="text" value="NULL"/> | い | <input type="text" value="NULL"/> | う | <input type="text" value="'\0'"/> |
| (E). | あ | <input type="text" value="'\0'"/> | い | <input type="text" value="'\0'"/> | う | <input type="text" value="NULL"/> | (F). | あ | <input type="text" value="'\0'"/> | い | <input type="text" value="'\0'"/> | う | <input type="text" value="'\0'"/> |
| (G). | あ | <input type="text" value="'\0'"/> | い | <input type="text" value="NULL"/> | う | <input type="text" value="NULL"/> | (H). | あ | <input type="text" value="'\0'"/> | い | <input type="text" value="NULL"/> | う | <input type="text" value="'\0'"/> |

(xi) 次のプログラムの出力結果を下の選択肢から 1 つ選べ。

```
#include <stdio.h>

struct point {
    int x, y;
};

void boki(struct point p) {
    p.y += 3;
};

void goki(int a[]) {
    a[1] += 3;
}

int main(void) {
    struct point pt = { 1, 2 };
    int array[] = { 1, 2 };

    boki(pt);
    goki(array);
    printf("%d %d ", pt.x, pt.y);
    printf("%d %d\n", array[0], array[1]);
    return 0;
}
```

- (A). 1 2 1 2 (B). 5 1 2 1 (C). 1 5 1 2 (D). 5 1 5 1
(E). 2 1 2 1 (F). 1 2 1 5 (G). 2 1 5 1 (H). 1 5 1 5

(xii) 次のプログラムの出力結果を下の選択肢から 1 つ選べ。

```
#include <stdio.h>

struct point {
    int x, y;
};

struct point doki(struct point p) {
    p.x += 2;
    return p;
}

struct point* zuki(struct point* pp) {
    pp->x += 2;
    return pp;
}

int main(void) {
    struct point p1 = { 3, 4 }, p2, p3 = { 7, 8 };
    struct point *ptr;

    p2 = doki(p1);
    ptr = zuki(&p3);
    printf("%d %d %d %d ", p1, p2);
    printf("%d %d %d %d %n", p3, *ptr);
    return 0;
}
```

(A). 5 4 5 4 9 8 9 8

(B). 5 4 5 4 7 8 9 8

(C). 5 4 5 4 9 8 7 8

(D). 3 4 5 4 9 8 7 8

(E). 3 4 5 4 7 8 9 8

(F). 3 4 5 4 9 8 9 8

