

## 修了試験「C 言語基礎 1」(1-B 第 2 回) 解答例

\_\_\_\_\_/100

出席番号 \_\_\_\_\_

社員番号 \_\_\_\_\_

氏名 \_\_\_\_\_

### 【問題 1】 (各 1 点、計 4 点) (P. 62・99)

以下の式の評価結果は真か偽か答えなさい。ただし a=15、b=16、c=20、d=20 とする。

- |     |                              |          |
|-----|------------------------------|----------|
| (1) | $(a \% 5) == (b \% 4)$       | <u>真</u> |
| (2) | $a++ == b$                   | <u>偽</u> |
| (3) | $(a == b) \    \ (c == d)$   | <u>真</u> |
| (4) | $(a == b) \ \&\& \ (c == d)$ | <u>偽</u> |

### 【問題 2】 記憶クラス指定子 (計 3 点) (C 言語 本格入門 P. 92)

#### const とは何か

初期化した値から一切変更できなくなる、記憶クラス指定子と言います。  
あらかじめ決められた値をプログラム中で変更したくない場合に利用します。  
複数のメンバーで開発する時、他のプログラマへ  
この変数は読み取り専用だと伝えることも出来ます。

下記に示すプログラムには、2 箇所の問題があります。

問題の箇所を文章で記しなさい。

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

```
int parallelogram(const int, const int);  
const double pai = 3.14159265;
```

```
int main()  
{
```

```
    int hankei = 5, height, base;
```

```
    printf("半径が%d の円の面積は%f です。¥n", hankei, hankei * hankei * pai);
```

```
printf("平行四辺形を求める\n");
printf("平行四辺形の高さは");
scanf("%d", &height);
printf("平行四辺形の底辺は");
scanf("%d", &base);
parallelogram(height, base);
}

int parallelogram(const int height, const int base)
{
    height = 7;
    printf("高さ%d 底辺の長さ%d 平行四辺形の面積は %d\n",
           height, base + 1, base * height);
    return 0;
}
```

解答

1. parallelogram() 関数で height を変更しているため、コンパイルエラー。(2 点)
2. parallelogram() 関数で表示している底辺が計算結果と相違がある。(1 点)  
(base + 1 → base) ※const している base に積算は行われていないため、  
コンパイルエラーは発生しない。

【問題 3】 (各 2 点、計 12 点) (P. 117・156)

以下のプログラムは、関数 mystrcat() により 2 つの文字列を連結するプログラムです。  
以下のプログラムへ穴埋めをしてください。

<プログラム>

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

void mystrcat(char *to, char *from);

int main(void)
{
    char str[80];

    strcpy(str, "最初の部分");
```

```
mystrcat(str, " 2 番目の部分");  
printf(str);  
return 0;  
}  
  
void mystrcat(char *to, char *from)  
{  
    /* to の最後を探す */  
    while( (1) != '¥0' )  
    {  
        (2)  
    }  
  
    /* to の後ろへ from の文字列を連結する */  
    while ( (3) != '¥0' )  
    {  
        (4) = (5)  
    }  
  
    /* 終端文字を追加する */  
    (6) = '¥0';  
}
```

(1) \*to

(2) to++;

(3) \*from

(4) \*to++

(5) \*from++;

(6) \*to

【問題 4】（各 2 点、計 8 点） (P. 89・175)

以下のプログラムを完成させなさい。

<プログラム概要>

数字の 0 から 9 を意味する英単語 (zero、one、two、…、nine) を入れた文字列テーブルを作成後、ユーザに (文字として) 数字を入力させ、このテーブルを使って該当する単語を表示する。

<プログラム>

```
#include <stdio.h>

int main(void)
{
    char digits[ ](1) = {
        "zero", "one", "two", "three",
        "four", "five", "six", "seven",
        "eight", "nine"
    };

    char num;

    printf("数字を入力してください：");
    num = getchar();
    printf("\n");

    num = num - (2);
    if ((3) && num < 10)
        printf("%s", (4));

    return 0;
}
```

(1) [10][10]

※又は [ ][10]

(3) num >= 0

(2) '0'

※又は、0x30, 48

(4) digits[num]

**【問題 5】**（各 1 点、計 6 点） **(P. 110・154)**

以下のプログラムには、記述ミスがあり動作しません。

プログラムを読んで記述ミスの行番号と正しい記述を解答欄に書きなさい。

＜プログラム 1＞

```
01    include <stdio.h>
02    int main(void)
03    {
04        char ch1[][10] = {"0123456789", "9876543210"};
05        char ch2[256];
06        char *cp;
07        int i;
08
09        for(i = 0; i <= 2; i++)
10        {
11            printf("%s¥n", &ch1[i][0]);
12        }
13
14        &cp = "abc";
15        printf("%s¥n", cp);
16        &cp = ch2;
17        scanf("%s", &cp);
18        printf("%s¥n", cp);
19
20        scanf("%s", ch2);
21        printf("%s", ch2);
22        return 0;
23    }
```

＜解答＞

```
01 #include <stdio.h>
04 char ch1[][11]
09 for(i = 0; i < 2; i++)
14 cp = "abc";
16 cp = ch2;
17 scanf("%s", cp);
```

【問題 6】（各 1 点、計 6 点） (P. 89・92)

以下に示すプログラムは 2 つの整数値を読み込んで、どちらが大きいか、または同じかを判定するプログラムです。

以下のプログラムへ穴埋めをしてください。

<プログラム>

```
#include <stdio.h>

int main(void)
{
    int na;                // 整数 A
    int nb;                // 整数 B

    printf("2 つの整数を入力して下さい。¥n");
    printf("整数 A:");
    (1) ("d", (2));
    printf("整数 B:");
    (1) ("d", (3));

    if ((4))
    {
        printf("A と B は等しい");
    }
    (5) ((6))
    {
        printf("A は B より大きい");
    } else
    {
        printf("A は B より小さい");
    }

    return 0;
}
```

(1) scanf

(2) &na

(3) &nb

(4) na == nb

(5) else if

(6) na > nb

【問題 7】 (各 1 点、計 6 点) (P. 65・89)

以下のプログラムは入力した数値が奇数か偶数か 10 の倍数かを判定するプログラムです。

以下のプログラムへ穴埋めをしてください。

<プログラム>

```
#include <stdio.h>
int main(void)
{
    int data;

    printf("整数を入力してください。¥n");
    scanf("(1)", (2));

    if(( (3) ) != 0)
    {
        printf("この数値は奇数です。¥n");
    }
    (4)
    {
        printf("この数値は偶数です。¥n");
    }

    if(( (5) ) == 0)
    {
        printf("この数値は 10 の倍数です。¥n");
    }

    (6)
}
```

(1)   %d    
(2)   &data    
(3)   data % 2  

(4)   else    
(5)   data % 10    
(6)   return 0;

以降の問題は実際にプログラムを行い、プログラムソース、実行ファイルは指定した場所に保存してください。

【問題 8】 (10 点) (P. 51・92)

scanf() を使って、自分の名前と生年月日(mm/dd/yy の書式で)と電話番号を入力するプログラムを作成してください。正しく入力された事を確認する為に、画面に情報を再表示するものとする。

```
#include <stdio.h>
int main(void)
{
    char name[80], date[80], phone[80];

    printf("名前と生年月日");
    printf("と電話番号を入力してください:¥n");
    scanf("%s%8s%8s", name, date, phone);
    printf("%s %s %s", name, date, phone);

    return 0;
}
```



【問題 9】 (10 点) (P. 110-175)

以下のような 2 次元配列を用意し、その全てを足した値を表示するプログラムを作成せよ。

ただし、操作は全て配列の添え字を使って行いなさい。

配列名 : data

配列内容 : { {1, 2, 3, 4, 5},  
              {6, 7, 8, 9, 10},  
              {99, 98, 97, 96, 95}}

<実行結果>

2 次元配列合計 : 540

```
#include <stdio.h>
#define DATA_SIZE1 5
#define DATA_SIZE2 3
int main(void)
{
    int data[DATA_SIZE2][DATA_SIZE1] = {
        {1, 2, 3, 4, 5},
        {6, 7, 8, 9, 10},
        {99, 98, 97, 96, 95}};

    int i, j;
    int sum = 0;

    for(i = 0; i < DATA_SIZE2; i++)
    {
        for(j = 0; j < DATA_SIZE1; j++)
        {
            sum = sum + data[i][j];
        }
    }

    printf("2 次元配列合計 : %d\n", sum);

    return 0;
}
```

**【問題 10】 (10 点) (P. 110・154・175)**

問題 9 と同じ条件で、全ての操作においてポインタを使って行うプログラムを作成してください。

```
#include <stdio.h>
#define DATA_SIZE1 5
#define DATA_SIZE2 3
int main(void)
{
    int data[DATA_SIZE2][DATA_SIZE1] = {
        {1, 2, 3, 4, 5},
        {6, 7, 8, 9, 10},
        {99, 98, 97, 96, 95}};
    int i;
    int *p = &data[0][0];
    int sum = 0;

    for(i = 0; i < (DATA_SIZE1 * DATA_SIZE2); i++)
    {
        sum = sum + *p++;
    }
    printf("2次元配列合計 : %d\n", sum);

    return 0;
}
```

**【問題 11】 (10 点) (P. 106・156)**

gets() 関数と同じ動作をする mygets() 関数を作成しなさい。ただし、下記の条件を満たすこと。

- ・ mygets() 関数内で標準入力に使用して良いのは getchar() 関数のみ。
- ・ mygets() 関数を呼び出し、入力された文字列を表示する main() 関数を作り、検証できるようにすること。
- ・ mygets() 関数の戻り値は考えなくて良いものとする。

```
#include <stdio.h>
void mygets(char *strP);

int main(void)
{
```

```
    char str[256];

    mygets(str);
    printf("%s\n", str);
    return 0;
}

void mygets(char *strP)
{
    while((*strP = getchar()) != '\n')
    {
        strP++;
    }
    *strP = '\0';
}
```

【問題 1 2】 (15 点) (P. 89・92・122)

試験の得点を入力し、入力点数により“優”、“良”、“可”、“不可”に分類し表示するプログラムを、下記の条件に従って作成してください。

条件：

- ・ 分類する人数を入力させ、その人数分に分類できること。ただし最大 100 人までとする。  
「評価する人数の入力」に 100 以上が入力された場合、再度入力を求める。
- ・ 分類は 90 点以上が“優”、70 点以上 89 点以下が“良”、50 点以上 69 点以下が“可”、49 点以下が“不可”とする。
- ・ 点数は 100 点満点とする。100 点以上の入力があった場合、再度入力を求める。

＜実行結果＞

評価する人数の入力：5

1 人目 得点：20

2 人目 得点：100

3 人目 得点：95

4 人目 得点：80

5 人目 得点：75

優 2 人

良 2 人

可 0 人

不可 1 人

```
#include <stdio.h>
int main(void)
{
    int man;
    int score;
    int yuu, ryou, ka, fuka;
    int i;

    yuu = ryou = ka = fuka = 0;
    do
    {
        printf("評価する人数の入力：");
        scanf("%d", &man);
    } while ((man <= 0) || (man > 100));

    i = 0;
```

```
while(i < man)
{
    printf("%d 人目 得点 :", i+1);
    scanf("%d", &score);
    if((score < 0) || (score > 100))
    {
        continue;
    }else if(score >= 90)
    {
        yuu++;
    }else if(score >= 70)
    {
        ryou++;
    }else if(score >= 50)
    {
        ka++;
    }else
    {
        fuka++;
    }
    i++;
}
printf("優 %d 人¥n 良%d 人¥n 可%d 人¥n 不可%d 人¥n", yuu, ryou, ka, fuka);

return 0;
}
```