

2015 年度 プログラミング II レポート 05

学生用

学籍番号：_____

氏名：_____

下記の注意事項を守り、次ページ以降の問いに答え、レポートを完成させなさい。

提出期限：2015 年 11 月 10 日 (火) 13:00 まで

提出場所：理学部棟 正面玄関内に設置のレポートボックス

注意事項：

- (1) このページを印刷し、必要事項を記入の上 (学籍番号欄と氏名欄は2箇所あるので忘れずに記入すること)、レポートの表紙として提出すること。
- (2) コンピュータ端末室を利用する場合は、情報システム利用ガイドラインを厳守すること。**特に、コンピュータ端末室では飲食禁止である。**
- (3) クラスメイトのレポートを参考にしたり、クラスメイトと協力してレポートを作成した場合は、教員控の協力者氏名欄にクラスメイトの氏名を記入すること。これらの場合も、自分の言葉で表現し直すこと。**コピー禁止。**
- (4) プログラミング II について、あなたの声を聞かせてください (教員控の意見・質問欄に記入のこと)。気軽にどうぞ (成績には一切影響しません)。

出題者：幸山 直人

出題日：2015 年 11 月 4 日 (水)

----- 切り取り線 -----

2015 年度 プログラミング II レポート 05

教員控

学籍番号：_____

氏名：_____

協力者氏名：_____, _____, _____

レポート作成に要した時間：_____. _____ 時間

意見・質問：

問 1 p.120 の記述にしたがって、ソースプログラム「平均を求める」(rei4_1a.c)を作成しなさい。さらに、ソースプログラムをコンパイルし、実行可能ファイルが正しく動作するか確認しなさい。なお、作成したソースプログラムは印刷してレポートに添付すること。
注意：コピー禁止。自らの手で全てのソースプログラムを入力すること。

解答例 著作権保護のため解答を記述していません。配布済みのファイルを参照してください。

問 2 p.121 の記述にしたがって、ソースプログラム「平均を求める」(rei4_1b.c)を作成しなさい。さらに、ソースプログラムをコンパイルし、実行可能ファイルが正しく動作するか確認しなさい。なお、作成したソースプログラムは印刷してレポートに添付すること。

解答例 著作権保護のため解答を記述していません。配布済みのファイルを参照してください。

問 3 以下のソースプログラム「平均を求める」(rei4_1c.c)は、p.123 を参考に問 2 のソースプログラム「平均を求める」(rei4_1b.c)の while 文を do ~ while 文に書き換えたものである。正しく動作するように第 24 行と第 25 行に適当な文を補ってソースプログラムを完成しなさい。さらに、ソースプログラムをコンパイルし、実行可能ファイルが正しく動作するか確認しなさい。なお、作成したソースプログラムは印刷してレポートに添付すること。

● 平均を求める

rei4_1c.c

```
： ----- 第 1 行から第 14 行までは rei4_1b.c と同じなので省略 -----  
  
15:    //入力と合計の計算  
16:    total = 0;        //初期設定  
17:    i = 0;  
18:    do {  
19:        printf("%d 科目目の得点を入力：", i + 1);  
20:        scanf("%d", &score);        //入力  
21:        total += score;            //合計を求める  
22:        i++;                    //1 回カウント  
23:    } while (score > END);  
24:                                //変数 total の値を修正  
25:                                //変数 i の値を修正  
26:    ave = (double)total / i;        //平均を求める  
27:  
28:    //表示  
29:    printf("%d 科目の合計：%5d 点 平均：%7.11f 点 ¥n", i, total, ave);  
30:  
31:    return 0;  
32: }
```

解答例 別紙を参照のこと。

問 4 p.125 の記述にしたがって、ソースプログラム「平均を求める」(rei4.2.c)を作成しなさい。さらに、ソースプログラムをコンパイルし、実行可能ファイルが正しく動作するか確認しなさい。なお、作成したソースプログラムは印刷してレポートに添付すること。

注意：for 文の式 1 の `total = 0` は繰り返しの変数に直接関係しないので、for 文の外に記述すること。

解答例 著作権保護のため解答を記述していません。配布済みのファイルを参照してください。
なお、ソースプログラム「平均を求める」(rei4.2.k.c)は注意事項にしたがって記述されたソースプログラムである。

参考：p.128 の記述にしたがって、ソースプログラム「平均を求める」(rei4.3a.c)を作成しなさい。さらに、ソースプログラムをコンパイルし、実行可能ファイルが正しく動作するか確認しなさい。なお、ソースプログラム「平均を求める」(rei4.3a.k.c)は注意事項にしたがって記述されたソースプログラムである。

問 5 p.131 の記述にしたがって、ソースプログラム「平均から評価を求める」(rei4.4a.c)を作成しなさい。さらに、ソースプログラムをコンパイルし、実行可能ファイルが正しく動作するか確認しなさい。なお、作成したソースプログラムは印刷してレポートに添付すること。

注意：コピー禁止。自らの手で全てのソースプログラムを入力すること。

解答例 著作権保護のため解答を記述していません。配布済みのファイルを参照してください。

参考：p.132 の記述にしたがって、ソースプログラム「平均から評価を求める」(rei4.4b.c)を作成しなさい。さらに、ソースプログラムをコンパイルし、実行可能ファイルが正しく動作するか確認しなさい。

問 6 p.135 の記述にしたがって、ソースプログラム「平均から評価を求める」(rei4.5a.c)を作成しなさい。さらに、ソースプログラムをコンパイルし、実行可能ファイルが正しく動作するか確認しなさい。なお、作成したソースプログラムは印刷してレポートに添付すること。

注意：コピー禁止。自らの手で全てのソースプログラムを入力すること。

解答例 著作権保護のため解答を記述していません。配布済みのファイルを参照してください。

問 7 p.140 の記述にしたがって、ソースプログラム「平均から評価を求める」(rei4.6a.c)を作成しなさい。さらに、ソースプログラムをコンパイルし、実行可能ファイルが正しく動作するか確認しなさい。なお、作成したソースプログラムは印刷してレポートに添付すること。

注意：コピー禁止。自らの手で全てのソースプログラムを入力すること。

解答例 著作権保護のため解答を記述していません。配布済みのファイルを参照してください。

問 8 p.142 の記述にしたがって、ソースプログラム「成績処理」(rei4_7a.c)を作成しなさい。さらに、ソースプログラムをコンパイルし、実行可能ファイルが正しく動作するか確認しなさい。なお、作成したソースプログラムは印刷してレポートに添付すること。
注意：コピー禁止。自らの手で全てのソースプログラムを入力すること。

解答例 著作権保護のため解答を記述していません。配布済みのファイルを参照してください。

問 9 p.146 の記述にしたがって、ソースプログラム「最高点を求める」(rei4_8a.c)を作成しなさい。さらに、ソースプログラムをコンパイルし、実行可能ファイルが正しく動作するか確認しなさい。なお、作成したソースプログラムは印刷してレポートに添付すること。

注意：コピー禁止。自らの手で全てのソースプログラムを入力すること。

注意：for 文の式 1 の `max = score[0]` は繰り返しの変数に直接関係しないので、for 文の外に記述すること。

解答例 著作権保護のため解答を記述していません。配布済みのファイルを参照してください。なお、ソースプログラム「平均を求める」(rei4_8a_k.c) は注意事項にしたがって記述されたソースプログラムである。

問 10 以下の実行結果と同じになるように、制御文を使ってソースプログラム「掛け算九九の表」(report05_01.c)を作成しなさい。さらに、ソースプログラムをコンパイルし、実行可能ファイルが正しく動作するか確認しなさい。なお、作成したソースプログラムは印刷してレポートに添付すること。

ヒント：ソースプログラム「成績処理」(rei4_7a.c)を参考に、for 文を 2 つ使った 2 段階の入れ子構造になるようにする。なお、下三角の部分は if 文を使って空白を出力する。

```
Z:¥src>report05_01.exe Enter
 1  2  3  4  5  6  7  8  9
   4  6  8 10 12 14 16 18
    9 12 15 18 21 24 27
   16 20 24 28 32 36
   25 30 35 36 40
   36 42 48 54
   49 56 63
   64 72
   81
```

```
Z:¥src>
```

考察：ソースプログラム「掛け算九九の表」(report05_01.c)に習って、ソースプログラム「16進数の掛け算表」(report05_01a.c)を作成してみましょう。

解答例 別紙を参照のこと。

問 11 以下の実行結果と同じになるように、ユークリッドの互除法を用いて、2つの正の整数 $x = 1234$, $y = 56$ の最大公約数 (Greatest Common Divisor; GCD) を求めるソースプログラム「最大公約数」(report05_02.c) を作成しなさい。さらに、ソースプログラムをコンパイルし、実行可能ファイルが正しく動作するか確認しなさい。なお、作成したソースプログラムは印刷してレポートに添付すること。

ヒント: ソースプログラム「平均を求める」(rei4_1a.c) を参考に、**while** 文を使って条件を満たすまで処理を繰り返すようにする。なお、剰余 (余り) の計算には算術演算子「%」を用いる。

```
Z:\$src>report05_02.exe   
gcd(1234,56)=2  
  
Z:\$src>
```

解答例 別紙を参照のこと。

問 12 以下の実行結果と同じになるように、2つの 3×3 行列

$$X = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}, \quad Y = \begin{pmatrix} 9 & 8 & 7 \\ 6 & 5 & 4 \\ 3 & 2 & 1 \end{pmatrix}$$

に対して、積 XY を求めるソースプログラム「行列の積」(report05_03.c) を作成しなさい (for 文を 3 回使った 3 段階の入れ子構造になる)。さらに、ソースプログラムをコンパイルし、実行可能ファイルが正しく動作するか確認しなさい。なお、作成したソースプログラムは印刷してレポートに添付すること。

```
Z:\$src>report05_03.exe   
30.000 24.000 18.000  
84.000 69.000 54.000  
138.000 114.000 90.000  
  
Z:\$src>
```

考察: ソースプログラム「行列の積」(report05_03.c) を参考に、様々な行列の積を計算できるプログラムを考えてみましょう。例えば、下記の 4×3 行列 X と 3×5 行列 Y の積 XY を求めるソースプログラム「行列の積 (一般化)」(report05_03a.c) を作成してみましょう。

$$X = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \\ 10 & 11 & 12 \end{pmatrix}, \quad Y = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 6 & 7 & 8 & 9 & 10 \\ 11 & 12 & 13 & 14 & 15 \end{pmatrix}$$

解答例 別紙を参照のこと。

問 3 の解答例 「平均を求める」 (rei4_1c.c)

```
1:  /*****
2:      平均を求める
3:  *****/
4:  #include <stdio.h>
5:  #define END -1      //-1 が入力されたら終わり
6:
7:  int main(void)
8:  {
9:      //変数の宣言
10:     int     score;      //得点
11:     int     total;      //合計
12:     double  ave;        //平均点
13:     int     i;          //回数のカウント
14:
15:     //入力と合計の計算
16:     total = 0;          //初期設定
17:     i = 0;
18:     do {
19:         printf("%d 科目めの得点を入力：", i + 1);
20:         scanf("%d", &score);      //入力
21:         total += score;            //合計を求める
22:         i++;                        //1 回カウント
23:     } while (score > END);
24:     total = total - END;            //変数 total の値を修正
25:     i = i - 1;                      //変数 i の値を修正
26:     ave = (double)total / i;        //平均を求める
27:
28:     //表示
29:     printf("%d 科目の合計：%5d 点   平均：%7.1lf 点 ¥n", i, total, ave);
30:
31:     return 0;
32: }
```

参考：第 24 行は「total = total - score;」としておくと確実だが、良心的な数値が入力されるものと仮定すれば、解答例のような文や「total = total - (-1);」・「total = total + 1;」といった文が許されます。

参考：第 26 行において変数 i が 0 の場合 (いきなり -1 を入力した場合)、定義されない除算 (0 で割る) が発生します。このような状態を防ぐには「if (i != 0) ave = (double)total / i;」のような文を記述する必要があります (本来なら、このようなエラーを回避するプログラムを作成しなければなりません)。

問 10 の解答例 「掛け算九九の表」 (report05_01.c)

```
1: #include <stdio.h>
2: #define N 10
3:
4: int main(void)
5: {
6:     int i, j;
7:
8:     for (i = 1; i < N; i++) {
9:         for (j = 1; j < N; j++) {
10:             if (i <= j) {
11:                 printf("%3d", i * j);
12:             }
13:             else {
14:                 printf("   ");
15:             }
16:         }
17:         printf("\n");
18:     }
19:
20:     return 0;
21: }
```

問 11 の解答例 「最大公約数」 (report05_02.c)

```
1: #include <stdio.h>
2:
3: int main(void)
4: {
5:     int x = 1234, y = 56, tmp;
6:
7:     printf("gcd(%d,%d)=", x, y);
8:     while (y != 0) {
9:         tmp = x;
10:        x = y;
11:        y = tmp % x;
12:    }
13:    printf("%d\n", x);
14:
15:    return 0;
16: }
```

問 12 の解答例 「行列の積」(report05_03.c)

```
1: #include <stdio.h>
2: #define N 3
3:
4: int main(void)
5: {
6:     double x[N][N] = {{1, 2, 3},
7:                        {4, 5, 6},
8:                        {7, 8, 9}};
9:     double y[N][N] = {{9, 8, 7},
10:                       {6, 5, 4},
11:                       {3, 2, 1}};
12:     double tmp;
13:     int i, j, k;
14:
15:     for (i = 0; i < N; i++) {
16:         for (j = 0; j < N; j++) {
17:             tmp = 0;
18:             for (k = 0; k < N; k++) {
19:                 tmp = tmp + x[i][k] * y[k][j];
20:             }
21:             printf("%8.3f", tmp);
22:         }
23:         printf("\n");
24:     }
25:
26:     return 0;
27: }
```