

2005 年度 プログラミング演習 II レポート 4

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：2005 年 11 月 8 日 (火) 13:15 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) ~~文章処理ソフトウェアや図形処理ソフトウェア等を駆使してレポートを作成し (問→解答→問→解答→… の順になるように記述すること)、A4 サイズの用紙に印刷して提出すること (手書きは不可)。~~
- (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) プログラミング演習について、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：2005 年 11 月 2 日 (水) (11 月 3 日 (木) 修正)

得点：

/ 3

----- 切り取り線 -----

2005 年度 プログラミング演習 II レポート 4

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

得点：

/ 3

意見・質問：

問 1 演習 1 の (1) の変数 `x`, `y` を与えるとその最大公約数を返す関数「`gcd(x, y)`」を作成し、そのソースプログラムを印刷して提出しなさい。

解答例 変数 `x`, `y` を与えるとその最大公約数を返す関数「`gcd(x, y)`」

```
1: #include <stdio.h>
2:
3: int gcd(int x, int y);
4:
5: int main(void)
6: {
7:     int x = 672, y = 204;
8:
9:     printf("gcd(%d,%d)=%d\n", x, y, gcd(x, y));
10:
11:     return 0;
12: }
13:
14: int gcd(int x, int y)
15: {
16:     int tmp;
17:
18:     while (y != 0) {
19:         tmp = x % y;
20:         x = y;
21:         y = tmp;
22:     }
23:
24:     return x;
25: }
```

評価基準 解答例に準じたソースプログラムであれば 1 点。

ポイント : プロトタイプの宣言。関数の記述方法。

演習 1 の (2) の解答例 3 変数の最大公約数を求めるには `main()` 関数を以下のように変更する。

```
5: int main(void)
6: {
7:     int x = 672, y = 204, z = 124;
8:
9:     printf("gcd(%d,%d,%d)=%d\n", x, y, z, gcd(gcd(x, y), z));
10:
11:     return 0;
12: }
```

問 2 演習 1 の (3) の変数 `x`, `y` を与えるとその最小公倍数を返す関数「`lcm(x, y)`」を作成し、そのソースプログラムを印刷して提出しなさい。

解答例 変数 `x`, `y` を与えるとその最小公倍数を返す関数「`lcm(x, y)`」

```
1: #include <stdio.h>
2:
3: int gcd(int x, int y);
4: int lcm(int x, int y);
5:
6: int main(void)
7: {
8:     int x = 672, y = 204;
9:
10:    printf("lcm(%d,%d)=%d\n", x, y, lcm(x, y));
11:
12:    return 0;
13: }
14:
15: int gcd(int x, int y)
16: {
17:     int tmp;
18:
19:     while (y != 0) {
20:         tmp = x % y;
21:         x = y;
22:         y = tmp;
23:     }
24:
25:     return x;
26: }
27:
28: int lcm(int x, int y)
29: {
30:     return x * y / gcd(x, y);
31: }
```

評価基準 解答例に準じたソースプログラムであれば 1 点。

ポイント : プロトタイプの宣言。関数の記述方法。

演習 2 の (1) の解答

	呼び出す関数	戻り値
main() 関数	func(5) ↓	↑ 5 * func(4) = 120
func() 関数	func(4) ↓	↑ 4 * func(3) = 24
func() 関数	func(3) ↓	↑ 3 * func(2) = 6
func() 関数	func(2) ↓	↑ 2 * func(1) = 2
func() 関数	func(1) ↓	↑ 1 * func(0) = 1
func() 関数	func(0) →	→ 1 = 1

演習 2 の (2) の解答例 「n! を求めるプログラム」

```

1: #include <stdio.h>
2:
3: int main(void)
4: {
5:     int n = 5;
6:     int a = 1, i;
7:
8:     for (i = 1; i <= n; i++) {
9:         a = a * i;
10:    }
11:    printf("%d!=%d\n", n, a);
12:
13:    return 0;
14: }
```

以下のように記述することもできる。

```

1: #include <stdio.h>
2:
3: int main(void)
4: {
5:     int n = 5;
6:     int a = 1, i;
7:
8:     for (i = n; i > 0; i--) {
9:         a = a * i;
10:    }
11:    printf("%d!=%d\n", n, a);
12:
13:    return 0;
14: }
```

問 3 演習 2 の (3) の `while` 文を使って「 $n!$ を求めるプログラム」を作成し、そのソースプログラムを印刷して提出しなさい。

解答例 「 $n!$ を求めるプログラム」

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

以下のように記述することもできる。

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

評価基準 解答例に準じたソースプログラムであれば 1 点。

ポイント : `while` 文の記述方法。

演習 3 の解答例 「複素数の四則演算を行なうプログラム」

```
1: #include <stdio.h>
2:
3: double x, y; /* x + yi */ ← グローバル変数
4:
5: void add(double a, double b, double c, double d);
6: void sub(double a, double b, double c, double d);
7: void mul(double a, double b, double c, double d);
8: void div(double a, double b, double c, double d);
9:
10: int main(void)
11: {
12:     double a = 1.2, b = -0.5; /* a + bi */
13:     double c = -2.2, d = 6.8; /* c + di */
14:
15:     add(a, b, c, d);
16:     printf("((%f)+(%f)i)+((%f)+(%f)i)=(%f)+(%f)i", a, b, c, d, x, y);
17:     sub(a, b, c, d);
18:     printf("((%f)+(%f)i)-((%f)+(%f)i)=(%f)+(%f)i", a, b, c, d, x, y);
19:     mul(a, b, c, d);
20:     printf("((%f)+(%f)i)*((%f)+(%f)i)=(%f)+(%f)i", a, b, c, d, x, y);
21:     div(a, b, c, d);
22:     printf("((%f)+(%f)i)/((%f)+(%f)i)=(%f)+(%f)i", a, b, c, d, x, y);
23:
24:     return 0;
25: }
26:
27: void add(double a, double b, double c, double d)
28: {
29:     x = a + c;
30:     y = b + d;
31: }
32:
33: void sub(double a, double b, double c, double d)
34: {
35:     x = a - c;
36:     y = b - d;
37: }
38:
39: void mul(double a, double b, double c, double d)
40: {
41:     x = a * c - b * d;
42:     y = a * d + b * c;
43: }
44:
45: void div(double a, double b, double c, double d)
46: {
47:     double tmp;
48:
49:     tmp = c * c + d * d; /* tmp != 0 */
50:     x = (a * c + b * d) / tmp;
51:     y = (-1 * a * d + b * c) / tmp;
52: }
```

ポイント：プロトタイプの宣言。関数の記述方法。グローバル変数。