演習問題 13

問 1. 以前に作成した int mini(int x, int y) を呼び出して、4 つの整数の最小値を求める関数

int mini4(int x, int y, int z, int t)

を作成し、実行例のように、キーボードから入力した整数の最小値を出力するプログラムを作成せよ。

実行例 1

```
4つの整数を入力してください。
整数１;11
整数２;13
整数３;16
整数４;21
4値の最小値は、11です。
```

実行例 2

```
4つの整数を入力してください。
整数１;21
整数２;13
整数３;16
整数４;61
4値の最小値は、13です。
```

問 2. 再帰を用いないで、r から n までの正整数の和を求める関数

int sum(int n, int r)

を作成せよ。

実行例 1

```
正の整数1を入力：10
正の整数2を入力：20
10から20までの整数の和は165です。
```

実行例 2

```
正の整数1を入力：4
正の整数2を入力：11
4から11までの整数の和は60です。
```

問 3. 再帰を用いないで、n の階乗 (n!) を計算する関数

int factorial(int n)

を作成せよ。

実行例

```
正の整数を入力して下さい。
8
1!=1
2!=2
3!=6
4!=24
5!=120
6!=720
7!=5040
8!=40320
```

問 4. n コの異なるものから r コを選んで並べる順列 (permutation) の総数 ${}_n\mathrm{P}_r$ を求める関数

int P(int n, int r)

を作成せよ。動作を確認するための適切な main 関数も作成し、完成したプログラムを実行せよ。

【注】n コのものから r コを選んで並べる順列の数 ${}_n\mathrm{P}_r$ は、

$${}_n\mathrm{P}_r = n(n-1)(n-2)\cdots(n-r+1)$$

で与えられる。

実行例

```
整数を入力:7
P(7, 0) = 1
P(7, 1) = 7
P(7, 2) = 42
P(7, 3) = 210
P(7, 4) = 840
P(7, 5) = 2520
P(7, 6) = 5040
P(7, 7) = 5040
```

問 5. 2つの整数nとrが与えられたとき、2つの整数の最大公約数 (greatest common divisor) は、以下のようにして求める (ユークリッドのアルゴリズム)。

(1) nをrで割る。割り切れたなら、rが最大公約数である。
(2) nがrで割り切れない場合には、nをrで割った余り(n%r) を求める。これをrと置き、元のrをnとおいて、(1) に戻る。

2つの数の組を (n, r) と書くことにしよう。例として、910と21(910, 21) の最大公約数を求めよう。910を21で割ると7余る。そこで、今度は、21と7の組 (21, 7) について考える。21は7で割り切れるので、7は、21と7の最大公約数であるとともに、910と21の最大公約数でもある。

2つの数nとrの最大公約数を求める関数

int gcd(int n, int r)

を?を埋めてプログラムを完成させなさい。

プログラム

```
#include<stdio.h>

int gcd(int n, int r){
   int tmp;
   while(r>0){
      tmp=?;
      r=n%r;
      n=?;
   }
   return tmp;
}

int main(void){

   int n1, n2, m;

   printf("2つの整数を入力して下さい。\n");
   printf("整数1:"); scanf("%d", &n1);
   printf("整数2:"); scanf("%d", &n2);

   m = gcd(n1, n2);

   printf("%d と%d の最大公約数は%d です。
\n", n1, n2, m);

   return 0;
}
```

実行例

```
2つの整数を入力して下さい。
整数1:1804
整数2:328
1804と328の最大公約数は164です。

2つの整数を入力して下さい。
整数1:105
整数2:385
105と385の最大公約数は35です。

2つの整数を入力して下さい。
整数1:245
整数2:193
245と193の最大公約数は1です。
```

問 6. 問 4 で作成した、順列の総数を求める関数 int P(int n, int r) を用いて、n コから r コを選ぶ組み合わせの総数を求める関数 int combination(int n, int r) を作成せよ。動作を確認するための適切な main 関数も作成し、完成したプログラムを実行せよ。

【注】n コのものから r コを選ぶ組み合わせの総数 ${}_n\mathrm{C}_r$ は、

$$ {}_n\mathrm{C}_r = \frac{n(n-1)(n-2)\cdots(n-r+1)}{r(r-1)(r-2)\cdots 1} $$

で与えられる。

プログラム

```
#include<stdio.h>

int P(int n, int r){

   /*  ここに文を書きます。*/

}

int combination(int n, int r){

   /* ここに文を書きます。*/

}

int main(void){

   /*
       ここに文を書きます。
       演習問題 12 問 7 を参照のこと。
   */

   return 0;
}
```

実行例

```
正の整数を入力：9
   1
   1   1
   1   2   1
   1   3   3   1
   1   4   6   4   1
   1   5  10  10   5   1
   1   6  15  20  15   6   1
   1   7  21  35  35  21   7   1
   1   8  28  56  70  56  28   8   1
   1   9  36  84 126 126  84  36   9   1
```

問 7. 2 つの正整数 x, y が与えられたとき、この 2 つの数の最小公倍数 (least common multiple) は、以下のようにして求められる。

x と y の最大公約数 (greatest common divisor) を G とすると、x= aG, y = bG と書くことができる。このとき、最小公倍数 L は、L = abG で与えられる。

【例】33 と 55 の最小公倍数を求める。

この 2 つの数の最大公約数 G は 11 であるので、33=3 × 11, 55 = 11 × 5 と書ける。そこで、この 2 つの数の最小公倍数 L は、 L = 3 × 5 × 11 = 165 となる。

問 5 で作成した最大公約数を求める関数 int gcd(int n, int r) を利用して、最小公倍数を求める関数 int lcm(int x, int y) を作成し、2 値の最小公倍数を求めるプログラムを完成させよ。

プログラム

```
#include<stdio.h>

int gcd(int n, int r){

  /* ここに文を書きます。*/

}


int lcm(int x, int y){

  /* ここに文を書きます。*/
}

int main(void){

   int n1, n2, m;

   printf("2 つの整数を入力して下さい。\n");
   printf("整数 1:"); scanf("%d", &n1);
   printf("整数 2:"); scanf("%d", &n2);

   m = lcm(n1, n2);

   printf("%d と%d の最小公倍数は%d です。
\n", n1, n2, m);

   return 0;
}
```

実行例

```
2つの整数を入力して下さい。
整数1:24
整数2:56
24と56の最小公倍数は168です。
```