

組	番号	名前	点数

設問 1

①	②	③	④
⑤			⑥

設問 2

①	②	③	④	⑤

設問 3

①	②	③	④	⑤
⑥	⑦			

設問 4

①	②	③	④	⑤
⑥	⑦	⑧	⑨	⑩

設問 5

①	②	③	④	⑤	⑥
⑦		⑧		⑨	

設問 6

①	②	③
④	⑤	⑥

## 設問 1

println はそこで指定されている内容を入力して改行するものである。一方、print は内容を入力して改行しないものである。下記のプログラムそれぞれについて出力結果がどうなるか回答せよ。

下記のプログラムを実行すると、「hello」という文字列が  ① 回表示される。

```
int i=10;
while( i > 0 ){
    println( "hello" );
    i--;
}
```

下記のプログラムを実行すると、「fms」という文字列が  ② 回表示される。

```
int i=10;
while( i <= 100 ){
    i++;
    if( i % 5 == 0 ){
        println( "fms" );
    }
}
```

下記のプログラムを実行すると、「fms!」という文字列が  ③ 回表示される。

```
void sayFMS( int num ){
    int n=0;
    while( n<num ){
        println( "fms!" );
        n++;
    }
}

void setup(){
    for( int i=0; i<3; i++ ){
        sayFMS( i );
    }
}
```

下記のように表示されるように④を埋めよ。

```
1
2
6
24
120
```

プログラム

```
int num = 1;
for( int i=1; i<=5; i++ ){
    num =  ④ ;
    println( num );
}
```

下記出力結果を⑤に記入せよ。

```
for( int i=10; i<=15; i++ ){
    if( i%3 == 0 ){
        print( "Fizz" );
    } else if( i%5 == 0 ){
        print( "Buzz" );
    } else if( i%3 == 0 && i%5 == 0 ){
        print( "Fizz Buzz" );
    } else {
        print( i );
    }
    print( ", " );
}
```

下記プログラムで表示されるのは  ⑥ である。

```
int [] values = new int [5];
for( int i=0; i<5; i++ ){
    values[i] = (i * 3) % 5;
}
for( int i=0; i<4; i++ ){
    for( int j=0; j<4; j++ ){
        if( values[j] < values[j+1] ){
            int temp = values[j];
            values[j] = values[j+1];
            values[j+1] = temp;
        }
    }
}
println( values[2] );
```

## 設問 2

Processing の String 型を使った下記の結果がどう出力されるか回答せよ。なお、解答欄の①～⑤は出力結果のそれぞれの行数に対応する。ただし、プログラム内で利用されているメソッドはそれぞれ下記のように定義されている。

### Name: length()

Returns the total number of characters included in the String as an integer number.

### Name: charAt()

Returns the character at the specified index. An index ranges from 0 to the length of the string minus 1. The first character of the sequence is at index 0, the next at index 1, etc.

### Name: toLowerCase()

Converts all of the characters in the string to lowercase. For example, "ABC" will convert to "abc".

### Name: toUpperCase()

Converts all of the characters in the string to uppercase. For example, "abc" will convert to "ABC".

### Name: indexOf()

Tests to see if a substring is embedded in a String, and returns the index position of the first occurrence of the substring defined in the str parameter. If the str substring is not found within the String, -1 is returned.

### Name: substring()

Returns a new string that is a part of the original string. When using the endIndex parameter, the string between beginIndex and endIndex-1 is returned.

str.substring(beginIndex)

str.substring(beginIndex, endIndex)

## プログラム 1

```
String str = "Frontier_Media_Science_(FMS)";

// 1 行目
println( "length=" + str.length() );

// 2 行目
println( str.charAt( 6 ) );

// 3 行目
println( str.indexOf( "fms" ) );

String str2 = str.replace( "M", "N" );
// 4 行目
println( str2.replace( "_", ", " ) );

int x = str2.indexOf( "(" ) + 1;
int y = str2.indexOf( ")" );

// 5 行目
println( str2.substring(x,y).toLowerCase() );
```

### 設問 3

コラッツ予想とは、下記のルールに従うとすべての自然数が最終的に1になるのではという予想である。ルールは下記のとおり。

- ・ ある数が奇数なら3を掛けて1を足す
- ・ ある数が偶数なら2で割る
- ・ 計算結果が1になるまで上記計算を繰り返す

2 から 1000 までの数について、その数と全てのステップを「[5]->16->8->4->2->1」のように表示するように①～⑤の穴を埋めよ。

また、完成した関数を利用して下記を行った時にどうなるか⑥～⑦について回答せよ。

OutputCollatz(6) のとき、3 行目の条件判定は⑥回行われる。また、OutputCollatz(6) の出力結果は⑦である。

### プログラム 2

```
1  ① OutputCollatz( int num ) {
2      print( "[" + num + "]" );
3      while ( ② ) {
4          if ( num % 2 == 1 ) {
5              num = ③;
6          } else {
7              num = ④;
8          }
9          print( "->" + num );
10     }
11     print( "¥n" );
12 }
13
14 void setup() {
15     int number = 2;
16     while ( number < ⑤ ) {
17         OutputCollatz( number );
18         number++;
19     }
20 }
```

## 設問 4

右記プログラムのように任意の丸（直径は 20）が赤色または白色で描画される OneLineLED クラスを作成した。

このOneLineLEDクラスでは、 ①  である OneLineLED( int \_num, int \_x, int \_y ) を利用することで、横に並ぶの丸の数と表示する XY 座標を指定することが可能となっている。また、この  ①  を用いて、下記メインプログラムでインスタンスを生成しているのはメインプログラムの  ②  行目である。

このクラスの shiftToLeft というメソッドを用いると、掲示板の赤丸が右から左へと 1 つずつ移動し、左端の赤丸が右端から登場する。一方、shiftToRight というメソッドを用いると、掲示板の赤丸が左から右へと 1 つずつ移動し、右端の赤丸が左端から登場する。

このクラスを完成させるため、設問中、プログラム中の丸囲み数字について回答せよ。

なおメインプログラムの 13 行目で標準出力される値は  ③  である。

メインプログラム

1	OneLineLED [] led = new OneLineLED [10];
2	for( int i=0; i<10; i++ ){
3	led[i] = new OneLineLED(20,20,i*20+10);
4	}
5	for( int j=0; j<5; j++ ){
6	for( int i=0; i<10; i++ ){
7	led[i].shiftToLeft();
8	}
9	}
10	for( int i=0; i<10; i++ ){
11	led[i].display();
12	}
13	println( led[5].lights[3] );

```
class OneLineLED {
    int [] lights;
    int num; // 横の数
    int x, y; //
    OneLineLED( int _num, int _x, int _y ) {
        num = _num;
        lights =  ④ ;
        for( int i=0; i<num; i++ ){
            lights[i] = i % 2;
        }
    }
    void display() {
        for( int i=0; i<num; i++ ){
            if ( lights[i] == 1 ) {
                fill( 255, 0, 0 ); // 赤色
            } else {
                fill( 255, 255, 255 ); // 白色
            }
            ellipse( i*20+x, y, 20, 20 );
        }
    }

    void shiftToLeft() {
        int i=0;
        int temp =  ⑤ ;
        while ( i < num-1 ) {
            lights[i] =  ⑥ ;
            i++;
        }
         ⑦  = temp;
    }

    void shiftToRight() {
        int i=num-1;
        int temp =  ⑧ ;
        while ( i > 0 ) {
            lights[i] =  ⑨ ;
            i--;
        }
         ⑩  = temp;
    }
}
```

## 設問 5

下記メインプログラムは、右記のクラス定義を利用したプログラムである。

ここで、Ball クラスは Object クラスを、Star クラスは Ball クラスを、Cross クラスは Ball クラスをそれぞれ  ①  したものである。メインプログラムの 17, 18, 19 行目に表示されるのは、下記の(ア)～(オ)のうちそれぞれ  ②  ,  ③  ,  ④  である。

(ア) ☆ (イ) ・ (ウ) ○ (エ) △ (オ) ×

また、17, 18, 19 行目に表示されるオブジェクトの中心座標 (x, y) は  ⑤  ,  ⑥  ,  ⑦  となる。

Object クラスで利用可能なインスタンス変数 (フィールド変数) は  ⑧  で、Cross クラスで利用可能なインスタンス (フィールド) 変数は  ⑨  である。設問中の①～⑨について回答せよ。

### メインプログラム

```
1 Ball b = new Ball();
2 Star st = new Star();
3 Cross cs = new Cross();
4 b.init( 50, 50, 10 );
5 cs.init( 100, 200, 5 );
6 st.init( 300, 200, 8 );
7 b.init( 100, 100, 20 );
8 int i=0;
9 while( i < 10 ){
10     b.move();
11     st.move();
12     if( i % 2 == 0 ){
13         cs.move();
14     }
15     i++;
16 }
17 b.display();
18 cs.display();
19 st.display();
```

### プログラム 5

```
class Object {
    int x;
    int y;
    int speed;
    Object() {
        x = 100;
        y = 100;
        speed = 0;
    }
    void init( int _x, int _y, int _s ){
        x = _x;
        y = _y;
        speed = _s;
    }
    void move() {
        x = x + speed;
    }
    void display() {
        point( x, y );
    }
}

class Ball extends Object {
    void move() {
    }
}

class Star extends Ball {
    void display() {
        ellipse( x, y, 30, 30 );
    }
}

class Cross extends Ball {
    int len;
    void init( int _x, int _y, int _s ) {
        len = 5;
    }
    void display() {
        line( x-len, y-len, x+len, y+len );
        line( x+len, y-len, x-len, y+len );
    }
}
```

## 設問 6

右記は、商品の金額を足して消費税 5%を含む合計金額を計算し、支払い額に対する釣り銭を表示する **Register** クラスと、それを用いるメインプログラムである。 **additem** メソッドで引数に指定した金額を加算し、 **sum** メソッドで合計金額を算出し、 **pay** メソッドで引数に指定の支払い金額に対する釣り銭を表示する。

メインプログラムの 9 行目が実行された時画面に表示されるのは、、12 行目が実行されたとき表示されるのは  である。

このプログラムは商品の登録を受け付けられなくなると **no more!** と表示するようになるものである。このメインプログラムで **no more!** と表示するには、メインプログラムの 13 行目の  の値を  以上にした時である。

なお、**Register** クラスを継承して、消費税 8% の場合の釣り銭を表示する **Register8** クラス、消費税 10% の場合の釣り銭を表示する **Register10** クラスを定義したい。どのように定義すれば良いか。④～⑥を埋めよ。

メインプログラム

```
1 void setup() {
2     Register regi;
3     regi = new Register(8);
4     for( int i=0; i<3; i++ ) {
5         regi.additem(100*i);
6     }
7     regi.additem(150);
8     regi.sum();
9     regi.pay(500);
10    regi.additem(50);
11    regi.sum();
12    regi.pay(1000);
13    for (int i=1; i<  ; i++) {
14        regi.additem(50);
15    }
16 }
```

Register8 クラス

```
class Register8 extends Register {
     {
        total =  ;
    }
}
```

Register10 クラス

```
class Register10 extends Register {
     {
        total =  ;
    }
}
```

Register クラス

```
class Register {
    int[] items;
    int numOfItems;
    int total;

    Register(int _num) {
        items = new int[_num];
        numOfItems = 0;
        total = 0;
    }

    void additem( int _price ) {
        if (numOfItems < items.length) {
            items[numOfItems] = _price;
            numOfItems ++;
            total = total + _price;
        } else {
            println("no more!");
        }
    }

    void sum() {
        total = (int) (total * 1.05);
    }

    void pay( int _payment) {
        int change = _payment - total;
        println("otsuri: " + change);
        total = 0;
        numOfItems = 0;
    }
}
```