

年	組	番号	名前	点数

設問 1

①	②	③	④	⑤

設問 2

①	②	③	④	⑤

設問 3

①	②	③	④	⑤
⑥	⑦	⑧	⑨	⑩

設問 4

①	②	③	④	⑤
⑥	⑦	⑧	⑨	⑩

設問 5

①	②	③	④	⑤

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出力して改行するものである。一方, print は内容を出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 1-1 を実行すると, 「fms」という文字列が 回表示される。

プログラム 1-1

```
void sayFMS( int num ){
    while( num > 0 ){
        println( "fms" );
        num--;
    }
}

void setup(){
    for( int i=5; i<8; i++ ){
        sayFMS( i );
    }
}
```

下記のように表示されるように②を埋めよ。

```
1
4
7
10
13
```

プログラム 1-2

```
for( int i=1; i<=15; i++ ){
    if(  ){
        println( i );
    }
}
```

下記のように表示されるように③を埋めよ。

```
1
5
9
13
17
21
```

プログラム 1-3

```
for( int i=1; i<=6; i++ ){
    println(  );
}
```

下記のように表示されるように④を埋めよ。

```
1
4
9
16
25
36
```

プログラム 1-4

```
for( int i=1; i<=6; i++ ){
    println(  );
}
```

下記プログラムで表示されるのは である。

プログラム 1-5

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

設問 2

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 2

```
String str = "ABCDEFGHJIJ";

println( str.length() );
println( str.indexOf("CDEF") );
println( str.indexOf("FMS") );

int x = str.indexOf( "C" );
int y = str.indexOf( "H" );

println( str.substring(x,y).toLowerCase() );

while( x < str.length() ){
    print( str.charAt( x ) );
    x++;
}
```

設問 3

右記プログラムは、配列の中にある目的の数値がどこにあるかを探す LinearSearch 関数と、配列の中にある目的の数値がいくつ含まれるかを数える CountKey 関数が含まれたプログラムである。

LinearSearch 関数に、配列と配列内での探索の開始位置、探したいデータを引数として渡すと、探索開始位置から最初に見つかる探したいデータの位置を、先頭から何番目にあるのかという値で返すものとなっている。

この関数では、渡された配列を先頭から順番に目的のデータと一致するか否かを検証していく。このアルゴリズムを線形探索という。

また、CountKey 関数に、配列と探したいデータを引数として渡すと、配列内にその探したいデータがいくつあるかを数えて数値で返すものとなっている。

プログラム中の①～⑦を埋めよ。

また、30 行目、31 行目、32 行目で出力される値は ⑧、⑨、⑩である。

プログラム 3

```
1 int LinearSearch(int[] d, int s, int k)
2 {
3     for(int i= ① ; i<d.length; i++){
4         if(d[i] == ② ){
5             return ③ ;
6         }
7     }
8     return ④ ;
9 }
10
11 int CountKey( int[] d, int k ){
12     int count = 0;
13     for( int i=0; i<d.length; i++ ){
14         if( ⑤ ) {
15             ⑥ ;
16         }
17     }
18     return ⑦ ;
19 }
20
21 void setup()
22 {
23     int [] data = new int [100];
24     for( int i=0; i<100; i++ ){
25         // 0 から 9 までの整数を階段状に設定
26         data[i] = i % 10;
27     }
28
29     println( LinearSearch( data, 0, 2 ) );
30     println( LinearSearch( data, 50, 2 ) );
31     println( CountKey( data, 2 ) );
32 }
33
```

設問 4

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。プログラムの3行目で Ball クラスの を利用し、 を作成している。

なお、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 Star クラスをそれぞれ 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 , , 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 , , となる。

Cross クラスで利用可能なインスタンス(フィールド)変数は である。

設問中の①~⑩について回答せよ。

メインプログラム

```
1 void setup(){
2     size( 400, 400 );
3     Ball ball = new Ball();
4     Star star = new Star();
5     Cross cross = new Cross();
6     ball.init( 50, 50, 10 );
7     star.init( 300, 200, 8 );
8     cross.init( 100, 200, 5 );
9     ball.init( 100, 100, 20 );
10    int i=0;
11    while( i < 10 ){
12        ball.move();
13        if( i % 2 == 0 ) cross.move();
14        i++;
15    }
16    star.move();
17    ball.display();
18    star.display();
19    cross.display();
20 }
```

プログラム 4

```
class Object {
    int x, 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 display(){
        ellipse( x, y, 30, 30 );
    }
}

class Star extends Ball {
    void move(){
        y = y + speed;
    }
}

class Cross extends Star {
    int len;
    void init( int _x, int _y, int _s ) {
        len = 5;
    }
    void display() {
        len = len + 1;
        line( x-len, y-len, x+len, y+len );
        line( x+len, y-len, x-len, y+len );
    }
}
```

設問 5

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掲示板クラスを作成した。

この掲示板クラスでは、コンストラクタである Keijiban(int h, int v)を利用することで、縦横の丸の数を指定することが可能となっている。

このクラスの shift 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。このクラスを用いて縦が 10 個、横が 20 個の掲示板を描画するインスタンスを作成したい。下記のプログラムの②を埋めることで実現せよ。また、プログラム中の丸囲み数字③～⑤についても回答することにより、問題なく動作するようにせよ。なお本プログラムを起動してから shift メソッドが 2 回呼び出された後、左上から右に 4 つ目、下に 3 つ目(lights[3][2])の○は①色になっている。

```
Keijiban myKeijiban = new Keijiban( ② );
```

プログラム 5

```
class Keijiban {
    int [][] lights;
    int numH; // 横の数
    int numV; // 縦の数

    Keijiban( int h, int v ) {
        numH = h;
        numV = v;
        lights = new int [numH][numV];
        for( int y=0; y<numV; y++ ) {
            for( int x=0; x<numH; x++ ){
                lights[x][y] = (x+y)%2;
            }
        }
    }

    void display() {
        for( int y=0; y<numV; y++ ) {
            for( int x=0; x<numH; x++ ){
                if ( lights[x][y] == 1 ) {
                    fill( 255, 0, 0 ); // 赤色
                } else {
                    fill( 255, 255, 255 ); // 白色
                }
                ellipse( x*20+10, y*20+10, 20, 20 );
            }
        }
    }

    void shift() {
        for( int y=0; y<numV; y++ ) {
            int temp = ③;
            for( int x=0; x<numH-1; x++ ){
                lights[x][y] = ④;
            }
            ⑤ = temp;
        }
    }
}
```

設問 6

右記プログラムは、関数 readData が返す値を画面上にグラフとして表示するプログラムである。関数 readData は、別に定義されている整数型の関数で、センサの観測値を 0~299 の範囲の整数で返す。

グラフは図 1 のように、最新の値を右端に表しながら、左へ移動するように表示される。これを実現するために、グラフの横幅と同じ大きさのデータを保存する整数配列 d を用意しデータを記録する。具体的には、図 2 のように、配列の末尾に readData() で取得されるデータを格納し、値を順に右から左へと移動することによって、600 フレーム分の情報を保存する。そのため、配列の先頭の値が 600 フレーム前(約 60 秒前)に readData() で取得されたデータとなる。

また、記録された全データを毎回描画することで、図 1 のようにグラフ全体が左へ移動するような表示を実現する。

新しいデータを毎秒 10 回読み込むこととするとき、プログラム中の空欄を埋めよ。

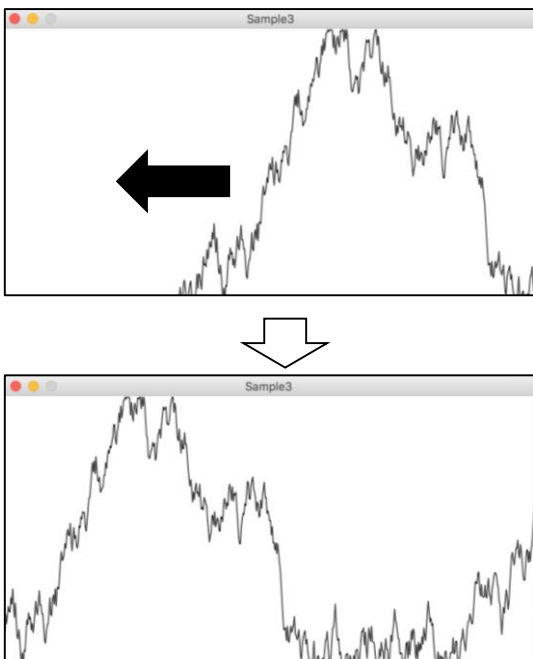


図 1

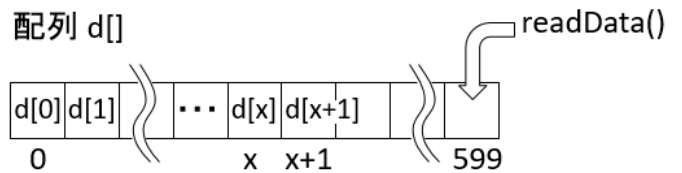


図 2

プログラム 6

```
int d[];

void setup() {
  size(600, 300);
  background(255);
  stroke(0);

  d = ; // 配列を用意する
  for( int x=0; x<width; x++ ){
    d[x] = 0; // 初期化する
  }
  frameRate( 10 ); // draw を毎秒 10 回に設定
}

void draw() {
  // 新しいデータを読み込んで記録する
  for( int x =  ; x  ; x  ){
    ;
  }
   = readData();

  // d に記録されているデータを表示する
  for( int x =  ; x  ; x  ){
    line(x, height-d[x], x+1, height-d[x+1]);
  }
}
```