

2014年10月27日 小テスト (3組用)

組	番	氏名	点数

設問 1

右記は、商品の金額を足して消費税 8%を含む合計金額を計算し、支払い額を指定すると釣銭を表示する Register クラスと、それを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した支払い金額に対する釣銭を表示する。メインプログラムでは、幾つかの商品の金額を登録し 1000 円支払った場合の釣り銭を表示している。

メインプログラム内の 2 行目では、Register クラスの を行っている。また、メインプログラムの 9 行目、11 行目が実行されたとき画面に表示されるのは、, である。

Register クラスを継承して、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。下記の , を埋めよ。

Register10 クラス

```
class Register10  {
    void sum() {
        total = total *  ;
    }
}
```

解答欄

①	インスタンス化
②	otsuri: 136
③	otsuri: 1000
④	extends Register
⑤	1.10

Register クラス

```
class Register {
    int total;
    Register() {
        init();
    }
    void init() {
        total = 0;
    }
    void additem( int _price ) {
        total = total + _price;
    }
    void sum() {
        total = total * 1.08;
    }
    void pay( int _payment ) {
        int change;
        change = _payment - total;
        println("otsuri: " + change);
        init();
    }
}
```

メインプログラム

```
1 void setup() {
2     Register regi = new Register();
3     for(int i=0; i<3; i++) {
4         regi.additem(200);
5     }
6     regi.additem(150);
7     regi.additem(50);
8     regi.sum();
9     regi.pay(1000);
10    regi.sum();
11    regi.pay(1000);
12 }
```

(裏へ続く)

設問 2

このページのプログラムは、四角形と三角形で構成される図形を移動、描画する Robot クラスと、Robot クラスを継承した Gonsuke クラスの定義、及びそれを使用するメインプログラムである。

Gonsuke クラスの 1 行目で Robot クラスを継承する記述は省略してある。Gonsuke クラスの中で、init メソッド、display メソッドは親クラスのメソッドを ⑥ している。メインプログラムの 5, 6 行目で Gonsuke, Robot クラスのインスタンス化をおこなう必要があるが ⑦ にはどのように記述すべきか記入せよ。このプログラムを実行すると、Gonsuke クラスの描くロボットは ⑨ 色の線で描画される。

この Gonsuke クラスは図 1 のような図形を表示させたいが、実際にはそのようにならない。正しく表示されるように Gonsuke クラスに 1 行だけ加筆して修正せよ (⑩)。

解答欄
⑥ オーバーライド
⑦ new Gonsuke()
⑧ new Robot()
⑨ 黒,
⑩ 回答はプログラム中に直接加筆せよ

Robot クラスと Gonsuke クラスの定義

```
class Robot {
  int x, y, sx, sy;
  int w = 40;
  int h = 40;
  Robot() {
    init();
  }
  void init() {
    x = width/2;
    y = height/2;
    sx = (int) random(1, 5);
```

```

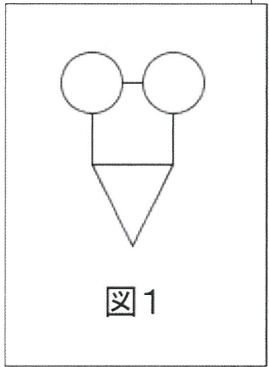
  sy = (int) random(1, 5);
  stroke(0);
  fill(255, 255, 255);
}

void move() {
  x = (x + sx) % width;
  y = (y + sy) % height;
}

void display() {
  rect(x, y, w, h);
  triangle(x, y+h, x+w/2, y+h*2, x+w, y+h);
}

class Gonsuke 【Robot クラスを継承】 {
  void init() {
    super.init();
    stroke(255, 0, 0);
    fill(255, 255, 255);
  }
  void display() {
    ellipse(x, y, 30, 30);
    ellipse(x+w, y, 30, 30);
  }
}

```



super.display(); を挿入

メインプログラム

```

1  Gonsuke gon;
2  Robot robo;
3  void setup() {
4    size(400, 400);
5    gon =  ⑦ ;
6    robo =  ⑧ ;
7  }
8  void draw() {
9    background(255);
10   robo.move();
11   robo.display();
12   gon.move();
13   gon.display();
14  }

```

2014年10月27日 小テスト (4組用)

組	番	氏名	点数

設問 1

右記は、商品の金額を足して消費税 8%を含む合計金額を計算し、支払い額に対する釣銭を表示する 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銭を表示する。

Register クラスのインスタンス化を行っているのは、メインプログラム内の ① 行目である。また、Register クラスで定義される total は、Register クラスの ② である。メインプログラムの 6, 9 行目が実行されたとき画面に表示されるのは、③、④ 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! と表示するが、メインプログラムの 10 行目の ⑤ がいくつ以上の時、no more! と表示するか。

解答欄	
①	2
②	インスタンス変数
③	otsuri: 284
④	otsuri: 946
⑤	10 11

メインプログラム

```

1 void setup() {
2   Register regi = new Register();
3   regi.additem(150);
4   regi.additem(50);
5   regi.sum();
6   regi.pay(500);
7   regi.additem(50);
8   regi.sum();
9   regi.pay(1000);
10  for (int i=0; i< ⑤ ; i++) {

```

```

11     regi.additem(50);
12   }
13 }

```

Register クラス

```

class Register {
  int[] items;
  int numOfItems;
  int total;
  Register(int _num) {
    init(_num);
  }
  Register() {
    init(10);
  }
  void init(int _num) {
    items = new int[_num];
    numOfItems = 0;
    total = 0;
  }
  void additem(int _price) {
    if (numOfItems < items.length) {
      items[numOfItems] = _price;
      numOfItems++;
    } else {
      println("no more!");
    }
  }
  void sum() {
    total = 0;
    for (int i=0; i<numOfItems; i++) {
      total = total + items[i];
    }
  }
  void pay(int _payment) {
    int change = _payment - (int)(total * 1.08);
    println("otsuri: " + change);
    total = 0;
    numOfItems = 0;
  }
}

```

(裏へ続く)

設問 2

このページのプログラムは、四角形と三角形で構成される図形を移動、描画する Robot クラスと、Robot クラスを継承した Gonsuke クラスの定義、及びそれを使用するメインプログラムである。

Gonsuke クラスが Robot クラスを継承するよう ⑥ を埋めよ。メインプログラムの 5, 6 行目で Robot, Gonsuke クラスの ⑦ をおこなっている。このプログラムを実行すると、Gonsuke クラスの描くロボットは ⑧ 色の線で描画され、Robot クラスの描くロボットは ⑨ 色の線で描画される。

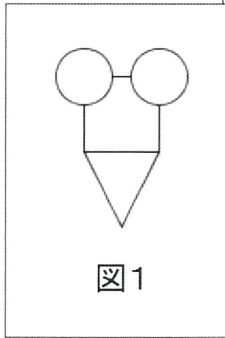
この Gonsuke クラスは図 1 のような図形を表示させたいが、実際にはそのようにならない。正しく表示されるように Gonsuke クラスを修正せよ(⑩)。

```

}
void move() {
    x = (x + sx) % width;
    y = (y + sy) % height;
}
void display() {
    rect(x, y, w, h);
    triangle(x, y+h, x+w/2, y+h*2, x+w, y+h);
}
}

class Gonsuke ⑥ {
    void init() {
        super.init();
        stroke(255, 0, 0);
        fill(255, 255, 255);
    }
    void display() {
        ellipse(x, y, 30, 30);
        ellipse(x+w, y, 30, 30);
        super.display();
    }
}

```



⑩ 1行目へ移動

解答欄	
⑥	extends Robot
⑦	インスタンス化
⑧	赤
⑨	赤
⑩	回答はプログラム中に直接加筆せよ

Robot クラスと Gonsuke クラスの定義

```

class Robot {
    int x, y, sx, sy;
    int w = 40;
    int h = 40;
    Robot() {
        init();
    }
    void init() {
        x = width/2;
        y = height/2;
        sx = (int) random(1, 5);
        sy = (int) random(1, 5);
        stroke(0);
        fill(255, 255, 255);
    }
}

```

メインプログラム

```

1 Robot robo;
2 Gonsuke gon;
3 void setup() {
4     size(400, 400);
5     robo = new Robot();
6     gon = new Gonsuke();
7 }
8 void draw() {
9     background(255);
10    gon.move();
11    gon.display();
12    robo.move();
13    robo.display();
14 }

```