

プログラミング A 第10回・宿題

宿題の提出は Moodle で月曜日までに行ってください。各宿題ごとに提出ファイルを zip 等で一つのファイルにまとめて該当する Moodle の課題に提出しなさい。この資料や関係するコードをインターネットなどに公開することは著作権上、禁止されています。

1 宿題 1

標準入力から整数の登録を受け付けて、平均や合計を求めるプログラムを作成して提出せよ。以下の全てを満足すること。ただし以降において N は任意の整数を表す。

- ">" を標準出力に出力し、利用者からの入力を受け付ける。
- 整数を追加登録したい場合は add N と入力する。例: add 100
- 登録済みの整数を削除したい場合は remove N と入力する。例: remove 100
- 登録済みの全ての整数を出力したい場合は list と入力し、カンマで区切られた形で一覧を標準出力に得る。出力の例 (100 と 50 を登録済みの状態で) : 100, 50,
- 登録済みの全ての整数を降順でソートしたい場合は sort と入力し、内部においてソートする。
- 登録済みの全ての整数の合計を出力したい場合は sum と入力し、合計を標準出力に得る。出力の例 (100 と 50 を登録済みの状態で) : 150
- 登録済みの全ての整数の算術平均を出力したい場合は mean と入力し、算術平均を標準出力に得る。出力の例 (100 と 50 を登録済みの状態で) : 75
- 終了したい場合は end と入力し、プログラムの実行を終了する。

例えば以下のように用いる。

```
> add 5
> add 1
> add 100
> add 70
> list
5,1,100,70,
> remove 100
> list
5,1,70,
> mean
25
> sum
76
> sort
> list
1,5,70,
> end
```

ヒント: 整数の集合を ArrayList や LinkedList で保持するとよい。ソートは java.util.Collections クラスの sort(List list) メソッドで簡単に実現できる。標準入力から得られたテキストの構文解析については java.util.Scanner クラスを用いると簡単である。

2 宿題 2

以下のそれぞれについて、この小テスト（宿題）や講義で扱うプログラムソースコードを参照する形で説明せよ。

- (1) 配列とコレクションの共通点と相違点
- (2) java.util.List と java.util.Set の共通点と相違点

3 宿題 3

「第 11 回 復習」(11_shukudai_fukushuu.pdf) を参考しながら下記の問題を回答せよ。

3.1 宿題 3-1

Study.java について、標準出力に表示する文字列を変更し、改変したプログラムソースコードの全体および実行結果を提出せよ。

参考資料: Study.java

```
public class Study {  
    public static void main(String[] args) {  
        System.out.println("こんにちは。");  
        System.out.println("私は ¥"Java¥" を勉強している。");  
    }  
}
```

3.2 宿題 3-2

Calc.java について、(100000L * 100000L) を (100000 * 100000) に変更して得られる実行結果、および、その結果を得た理由を回答欄に記述しなさい。

参考資料: Calc.java

```
public class Calc {  
    public static void main(String[] args) {  
        System.out.println("(7+8)/7=" + ((7 + 8) / 7));  
        System.out.println("(7+8)%7=" + ((7 + 8) % 7));  
        System.out.println("100000*100000=" + (100000L * 100000L));  
    }  
}
```

3.3 宿題 3-3

VarInt.java について、実行結果として 7.5 を標準出力に得るように一行のみを改変し、改変したプログラムソースコードの全体および実行結果を提出せよ。ただし、7.5 という数字そのものはプログラムソースコード中に記載しないこと（例えば System.out.println(7.5); などと書かないこと）。

参考資料：VarInt.java

```
public class VarInt {
    public static void main(String[] args) {
        int x = 7 + 8;
        int y = 2;
        y = x / y;
        System.out.println(y);
    }
}
```

3.4 宿題 3-4

VarDouble.java の内容を理解し、実行結果およびその結果を得た理由を回答欄に記述しなさい。

参考資料：VarDouble.java

```
public class VarDouble {
    public static void main(String[] args) {
        double x = 7 + 8;
        double y = 2.0;
        y = x / y;
        System.out.println(y + 314E-2);
    }
}
```

3.5 宿題 3-5

KeyboardInput.java について、キーボードから入力された単精度 32bit の浮動小数点数を文字列として読み込み、それを単精度 32bit の浮動小数点数に変換して標準出力に表示するように改変し、改変したプログラムソースコードの全体および実行結果（ソースコードの最後にコメントとして）を提出せよ。

参考資料：KeyboardInput.java

```
import java.io.*;
public class KeyboardInput {
    public static void main(String[] args) {
        BufferedReader reader = new BufferedReader(new InputStreamReader(System.in));
        try {
            String line = reader.readLine(); // 文字列の読み込み
            int number = Integer.parseInt(line); // 整数への変換
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

```
        System.out.println("入力された整数: " + number);
    } catch (Exception e) {
        System.out.println(e);
    }
}
}
```

3.6 宿題 3-6

IfStatement.java の内容を理解したうえで、x の値として適当なものを複数個用意して、それぞれの値と実行結果を参照しながらプログラムの仕組みを説明しなさい（例: プログラムは x の値が〇〇以上の場合に・・・となる。例えば x=999 の場合は・・・）。

参考資料: IfStatement.java

```
public class IfStatement {
    public static void main(String[] args) {
        int x = 60;
        if(x >= 0 && x <= 100) {
            if(x >= 80) {
                System.out.println("合格");
            } else {
                System.out.println("不合格");
            }
        } else {
            System.out.println("エラー");
        }
    }
}
```

3.7 宿題 3-7

SwitchStatement.java の内容を理解したうえで、answer の値として適当なものを複数個用意して、それぞれの値と実行結果を参照しながらプログラムの仕組みを説明しなさい（例: プログラムは answer の値が〇〇の場合に・・・となる。例えば・・・）。さらに、default を全て取り除いた場合にプログラムが示す挙動を説明しなさい。

参考資料: SwitchStatement.java

```
public class SwitchStatement {
    public static void main(String[] args) {
        char answer = 'n';
        switch(answer) {
            case 'y':
                System.out.println("Yes");
                break;
            case 'n':
```

```

        System.out.println("No");
        break;
    default:
        System.out.println("エラー");
        break;
    }
}
}

```

3.8 宿題 3-8

ForStatement.java について、for 文の代わりに while 文を用いて同じ処理を実現するように改変し、改変したプログラムソースコードの全体および実行結果提出せよ。

参考資料：ForStatement.java

```

c class ForStatement {
    public static void main(String[] args) {
        for (int i = 0; i < 10; i++) {
            System.out.print(i + ";"); //改行なし文字列表示
            for (int j = 0; j < i; j++) {
                System.out.print("*");
            }
            System.out.println("");
        }
    }
}

```

3.9 宿題 3-9

WhileStatement.java について、while 文の代わりに for 文を用いて同じ処理を実現するように改変し、改変したプログラムソースコードの全体を提出せよ。

参考資料：WhileStatement.java

```

import java.io.*;
public class WhileStatement {
    public static void main(String[] args) {
        BufferedReader reader = new BufferedReader(new InputStreamReader(System.in));
        try {
            String line;
            while ((line = reader.readLine()) != null) {
                System.out.println(line);
            }
        } catch (Exception e) {
            System.out.println(e);
        }
    }
}

```

```
}
```

3.10 宿題 3-10

WhileAndString.java について、キーボードから入力された文字列中のアルファベットを全て（小文字ではなく）大文字に変更して標準出力に表示するように改変し、改変したプログラムソースコードの全体を提出せよ。

参考資料：WhileAndString.java

```
import java.io.*;
public class WhileAndStatement {
    public static void main(String[] args) {
        BufferedReader reader = new BufferedReader(new InputStreamReader(System.in));
        try {
            String line;
            while ((line = reader.readLine()) != null) {
                s = line.toLowerCase();
                System.out.println(s);
            }
        } catch (Exception e) {
            System.out.println(e);
        }
    }
}
```

3.11 宿題 3-11

Method.java の内容を理解したうえで、getPower メソッドの呼び出しにおける引数 base, exponent として適当なものを複数組用意して、それぞれの値と実行結果を参照しながら getPower メソッドの推測される目的や仕組みを説明しなさい（例えば、0 の 0 乗はどのような値として得られるだろう？）。

参考資料：Method.java

```
public class Method {
    public static void main(String[] args) {
        System.out.println(getPower(2, 8));
    }
    public static int getPower(int base, int exponent) {
        int power = 1;
        for (int i = 0; i < exponent; i++) {
            power = power * base;
        }
        return power;
    }
}
```

3.12 宿題 3-12

Array.java について、配列中の全ての値の（合計ではなく）平均値、最小値、最大値をそれぞれ標準出力に表示するように改変し、改変したプログラムソースコードの全体を提出せよ。ただし平均値として相加平均を採用し、倍精度 64bit の浮動小数点数として表示すること。

参考資料：Array.java

```
public class Array {
    public static void main(String[] args) {
        int[] values = new int[]{ 62, 90, 75 };
        int sum = 0;
        for (int i = 0; i < values.length; i++) {
            sum += values[i];
        }
        System.out.println(sum);
    }
}
```

3.13 宿題 3-13

Array2.java について、配列中の全ての値の（合計ではなく）平均値、最小値、最大値をそれぞれ標準出力に表示するように改変し、改変したプログラムソースコードの全体を提出せよ。ただし平均値として相加平均を採用し、倍精度 64bit の浮動小数点数として表示すること。

参考資料：Array2.java

```
public class Array2 {
    public static void main(String[] args) {
        int[][] vs = new int[][]{
            { 62, 90, 75 },
            { 100, 0, 50 },
            { 30, 40 }
        };
        int sum = 0;
        for (int i = 0; i < vs.length; i++) {
            for (int j = 0; j < vs[i].length; j++){
                sum += vs[i][j];
            }
        }
        System.out.println(sum );
    }
}
```
