

# 電子計算機研究会

## 第四回オープン授業

### ループ

# 復習問題

今何時?と聞いてきた後24以下の数字を入力すると その時間を表示するプログラムを作成する(25以上の場合は『不正な時間です』と表示する)

## 実行例

```
<終了> IfReview [Java アプリケーション] /Library/Java/
今何時?
5
5時です
```

```
<終了> IfReview [Java アプリケーション] /Li
今何時?
27
不正な時間です
```

# 解答

```
package test;

import java.util.Scanner;

/**
 * 現在の時間を入力して判定
 * @author k.minamoto
 */
public class IfReview {
    public static void main(String[] args) {
        System.out.println("今何時?");
        Scanner scanner = new Scanner(System.in);
        int time = scanner.nextInt();
        if(time>24){
            System.out.println("不正な時間です");
        }else{
            System.out.println(time+"時です");
        }
    }
}
```

# for文

・ for(①初期設定; ②実行条件; ③継続処理) {

実行する文

}

1. ①を実行
2. ②かどうか判定、②を満たしていないならfor文終了
3. { }の間の繰り返したい文を実行
4. ③の処理を行い、再び②に戻る

# 例題1 (for文)

```
package test;
```

```
/**
```

```
 * for文
```

```
 */
```

```
public class For1 {
```

```
    public static void main(String[] args) {
```

```
        for(int i=0; i<5; i++){
```

```
            System.out.println(i);
```

```
        }
```

```
    }
```

```
}
```

1. int i=0;を実行(iに0を代入)
2. i<5かどうか判定し、iが5以上なら終了
3. System.out.println(i);でiの内容を表示して改行
4. iを+1する、2に戻る

# 実行結果

<終了> For1 [Java アプリケーション] /Library/Java/JavaVirtual

0

1

2

3

4

# 例題2{for文(入力あり)}

```
package test;

import java.util.Scanner;

/**
 * for文
 */
public class For2 {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("回数を入力してください");
        int num = scanner.nextInt();

        for(int i=0; i<num; i++){
            System.out.println("わーい!");
        }

        scanner.close();
    }
}
```

1. 回数を入力する
2. 入力した回数を変数numに代入する
3. numの回数だけfor文の中身を繰り返す

# 実行結果

<終了> For2 [Java アプリケーション] /Library/Java/JavaVirtualMachines/jdk

回数を入力してください10

わーい！

わーい！

わーい！

わーい！

わーい！

わーい！

わーい！

わーい！

わーい！

わーい！



# 例題3(二重for文)

```
package test;
```

```
/**
```

```
 * 二重for文
```

```
 */
```

```
public class DoubleFor1 {  
    public static void main(String[] args) {  
        for(int i=0; i<10; i++){  
            for(int j=0; j<i; j++){  
                System.out.print("*");  
            }  
            System.out.println();  
        }  
    }  
}
```

# 例題3解説

例題3では一つ目のfor文の間に二つ目のfor文を実行することで実装している。

```
for(int j=0; j<i; j++){  
    System.out.print("*");  
}  
System.out.println();
```

例題3は上記の文を10回実行しており、この文の1回目の実行ではiが0のため \* の表示がされない。 2回目の実行ではiが1に増えているため \* が一回表示される。これを繰り返すことによって、\*の個数が増えていく

# 実行結果

<終了> DoubleFor1 [Java アプリケーション] /Library/Java/JavaVirtual

---

\*

\*\*

\*\*\*

\*\*\*\*

\*\*\*\*\*

\*\*\*\*\*

\*\*\*\*\*

\*\*\*\*\*

\*\*\*\*\*

# While文

- ・ While(条件式) {  
    実行する文  
}
- ・ 条件式を満たす限り、{}の中身の文を実行し続ける

# 例題4

```
package test;

/**
 * while文
 */
public class While1 {
    public static void main(String[] args) {
        int i=0;

        while(i<10){
            i++;
            System.out.println(i);
        }
    }
}
```

- ・ 上記のwhileの例ではi<10の間、{ }の間の文を 実行している。

# 実行結果

<終了> While1 [Java アプリケーション] /Library/Java/JavaVirtualMach

---

1  
2  
3  
4  
5  
6  
7  
8  
9  
10

# break文

## for文の場合

```
・ for(①初期設定; ②実行条件; ③継続処理) {  
    実行する文  
    if(条件式) {  
        break;  
    }  
}
```

## While文の場合

```
・ while(条件式) {  
    実行する文  
    if(条件式) {  
        break;  
    }  
}
```

- ・ for文やWhile文のループを抜けるために使う

# 例題5

```
package test;

/**
 * 無限ループ
 */
public class EndlessRoop1 {
    public static void main(String[] args) {
        int i=0;

        for(;;){
            System.out.println("けもの");

            i++;

            if(i>=10){
                break;
            }
        }

        i=0;

        while(true){
            System.out.println("フレンズ");

            i++;

            if(i>=10){
                break;
            }
        }
    }
}
```



# 例題5解説

- ・ `i`が10以上になるまで『けもの』と表示する。その後`i`を0に戻し、また`i`が10以上になるまで『フレンド』と表示する。(iが10以上になったときに、`break`文が実行される)
- ・ `for(;;)`と`While(true)`は無限回実行を意味する。

# 実行結果

<終了> EndlessRoop1 [Java アプリケーション] /Library/Java

けもの

けもの

けもの

けもの

けもの

けもの

けもの

けもの

けもの

けもの

フレンズ

フレンズ

フレンズ

フレンズ

フレンズ

フレンズ

フレンズ

フレンズ

フレンズ

フレンズ

# 練習問題1

- ・ 1から100までの数字をすべて足した計算結果を表示するプログラムを作成する

# 練習問題1 解答

```
package test;

/**
 * for文
 */
public class ForQuestion {
    public static void main(String[] args) {
        int sum=0;

        for(int i=0; i<=100; i++){
            sum+=i;
        }

        System.out.println(sum);
    }
}
```

# 実行結果

<終了> ForQuestion [Java アプリケーション] /Library/Ja  
5050

# 練習問題2

- ・ 回数を入力した後に、その回数分の『一』が表示される、すっごーいという文章が表示されるプログラムを作成する。

## 実行例

(この実行例では『一』が5回表示される)

```
<終了> WhileQuestion [Java アプリケーション] /Library/Ja  
回数を入力してください5  
すっごー-----い！  
.....
```

# 練習問題2解答

```
package test;

import java.util.Scanner;

public class WhileQuestion {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("回数を入力してください");
        int num = scanner.nextInt();

        int i=0;

        System.out.print("すっご");

        while(i<num){
            System.out.print("—");

            i++;
        }

        System.out.print("い！");

        scanner.close();
    }
}
```

# 練習問題3

- 二重for文を使って以下のような実行結果を表示するプログラムを作成する

## 実行例

<終了> DoubleForQuestion [Java アプリケーション] /Librar

```
*
 *
  *
   *
    *
     *
      *
       *
        *
         *
          *
```



# 練習問題3解答

```
package test;

/**
 * 二重for文問題
 */
public class DoubleForQuestion {
    public static void main(String[] args) {
        for(int i=0; i<10; i++){
            for(int j=0; j<i; j++){
                System.out.print(" ");
            }
            System.out.println("*");
        }
    }
}
```

# 練習問題4

- ・ 前回の資料の練習問題2をfor文を使って書き直す

# 前回練習問題2

```
package test;

import java.util.Scanner;

public class Open_3_4_ans {

    public static void main(String[] args) {
        int time = 10;
        int life = 10;
        int distance = 10;
        Scanner scanner = new Scanner(System.in);

        //~~~~~ 1回目 ~~~~~

        System.out.println("残り時間：" + time + "、プレイヤーの体力：" + life + "、ゴールまでの距離：" + distance);
        System.out.println("どうする?( \"run\" or \"fight\" )");
        String behavior = scanner.nextLine(); // scanner.nextLine() で改行までの入力をbehaviorに代入
        if(behavior.equals("run")){ // この条件式は例題1を参照
            System.out.println("プレイヤーは走った!");
            System.out.println("敵の攻撃! プレイヤーに4ダメージ!");
            life -= 4; // life = life - 4;
            System.out.println("プレイヤーは4移動した!");
            distance -= 4; // distance = distance - 4;
        }
        else if(behavior.equals("fight")){// else がなくてもコンパイルは通る この条件式も例題1を参照
            System.out.println("プレイヤーの攻撃!\n敵を倒した!");
            System.out.println("プレイヤーは2移動した!");
            distance -= 2; // distance = distance - 2;
        }
        else{ //入力が無い間違えている時はただひたすら敵に殴られるのみ
            System.out.println("プレイヤーには何もできない!");
            System.out.println("敵の攻撃! プレイヤーに4ダメージ!");
            life -= 4; // life = life - 4;
        }
        time -= 4; // 時間は過ぎる
        System.out.println("- - - - -");
    }
}
```

# 前回練習問題2続き

//~~~~~ 2回目 ~~~~~

```
System.out.println("残り時間：" + time + "、プレイヤーの体力：" + life + "、ゴールまでの距離：" + distance);
System.out.println("どうする？( \"run\" or \"fight\" )");
behavior = scanner.nextLine();
if(behavior.equals("run")){
    System.out.println("プレイヤーは走った！");
    System.out.println("敵の攻撃！プレイヤーに4ダメージ！");
    life -= 4;
    System.out.println("プレイヤーは4移動した！");
    distance -= 4;
}
else if(behavior.equals("fight")){
    System.out.println("プレイヤーの攻撃！\n敵を倒した！");
    System.out.println("プレイヤーは2移動した！");
    distance -= 2;
}
else{
    System.out.println("プレイヤーには何もできない！");
    System.out.println("敵の攻撃！プレイヤーに4ダメージ！");
    life -= 4;
}
time -= 4;
System.out.println("- - - - -");
```

# 前回練習問題2続き

//~~~~~ 3回目 ~~~~~

```
System.out.println("残り時間：" + time + "、プレイヤーの体力：" + life + "、ゴールまでの距離：" + distance);
System.out.println("どうする？( \"run\" or \"fight\" )");
behavior = scanner.nextLine();
if(behavior.equals("run")){
    System.out.println("プレイヤーは走った！");
    System.out.println("敵の攻撃！プレイヤーに4ダメージ！");
    life -= 4;
    System.out.println("プレイヤーは4移動した！");
    distance -= 4;
}
else if(behavior.equals("fight")){
    System.out.println("プレイヤーの攻撃！\n敵を倒した！");
    System.out.println("プレイヤーは2移動した！");
    distance -= 2;
}
else{
    System.out.println("プレイヤーには何もできない！");
    System.out.println("敵の攻撃！プレイヤーに4ダメージ！");
    life -= 4;
}
time -= 4;
System.out.println("- - - - -");
```

//~~~~~ 結果 ~~~~~

```
if(life <= 0){
    System.out.println("プレイヤーは力尽きた...");
}
else if(distance <= 0){
    System.out.println("到着！ゲームクリア！！");
}
else if(time <= 0){
    System.out.println("間に合わなかったようだ...");
}
scanner.close();
```

}

}

# 練習問題4解答

```
package test;
import java.util.Scanner;

public class Open_3_4 {

    public static void main(String[] args) {
        int time = 10;
        int life = 10;
        int distance = 10;
        Scanner scanner = new Scanner(System.in);

        for(int i = 0; i < 3; i++) {
            System.out.println("残り時間:" + time + "、プレイヤーの体力:" + life + "、ゴールま
での距離:" + distance);
            System.out.println("どうする?( \"run\" or \"fight\" )");
            String behavior = scanner.nextLine();
            if(behavior.equals("run")) {
                System.out.println("プレイヤーは走った!");
                System.out.println("敵の攻撃!プレイヤーに4ダメージ!");
                life -= 4;
                System.out.println("プレイヤーは4移動した!");
                distance -= 4;
            }
        }
    }
}
```

# 練習問題4解答続き

```
else if(behavior.equals("fight")) {
    System.out.println("プレイヤーの攻撃!\n敵を倒した!");
    System.out.println("プレイヤーは2移動した!");
    distance -= 2;
}
else {
    System.out.println("プレイヤーは何もできない!");
    System.out.println("敵の攻撃!プレイヤーに4ダメージ!");
    life -= 4;
}
time -= 4;
System.out.println("- - - - -");
}

if(life <= 0) {
    System.out.println("プレイヤーは力尽きた_");
}
else if(distance <= 0) {
    System.out.println("到着!ゲームクリア!!");
}
else if(time <= 0) {
    System.out.println("間に合わなかったようだ...");
}
scanner.close();
}
}
```

# 練習問題5

- 100マス計算の表を表示するプログラムを作成する

## 実行例

```
<終了> RoopQuestion [Java アプリケーション] /Library/Java/JavaVirtualMachines/  
+---+---+---+---+---+---+---+---+---+---+---+  
| + | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |  
+---+---+---+---+---+---+---+---+---+---+---+  
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |  
+---+---+---+---+---+---+---+---+---+---+---+  
| 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |  
+---+---+---+---+---+---+---+---+---+---+---+  
| 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 |  
+---+---+---+---+---+---+---+---+---+---+---+  
| 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |  
+---+---+---+---+---+---+---+---+---+---+---+  
| 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |  
+---+---+---+---+---+---+---+---+---+---+---+  
| 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |  
+---+---+---+---+---+---+---+---+---+---+---+  
| 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 |  
+---+---+---+---+---+---+---+---+---+---+---+  
| 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 |  
+---+---+---+---+---+---+---+---+---+---+---+  
| 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 |  
+---+---+---+---+---+---+---+---+---+---+---+  
| 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |  
+---+---+---+---+---+---+---+---+---+---+---+
```



# 練習問題5解答

```
package test;

/**
 * 演習問題
 */
public class RoopQuestion {
    public static void main(String[] args) {
        System.out.println("+---+---+---+---+---+---+---+---+---+");
        System.out.println("| + | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |");
        for(int i=1; i<11; i++){
            System.out.println("+---+---+---+---+---+---+---+---+---+");
            if(i>9){
                System.out.print("| "+i+"|");
            }
            else{
                System.out.print("| "+i+" |");
            }
            for(int j=1; j<11; j++){
                if((i+j)>9){
                    System.out.print(" "+(i+j)+"|");
                }
                else{
                    System.out.print(" "+(i+j)+" |");
                }
            }
            System.out.println();
        }
        System.out.println("+---+---+---+---+---+---+---+---+---+");
    }
}
```