

点/100点

【問1】 次の各文の内容が正しい場合は「○」を、間違いの場合は「×」を答えなさい。 [各2計40]

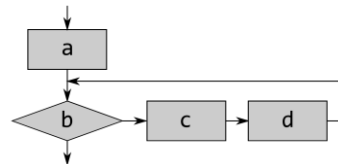
- 論理演算子!は「～ではない」の演算を行います
- 論理演算子!は単項演算子です
- 条件演算子?: は3項演算子です
- switch文は繰り返し文の1つです
- for文は前判定ループです
- do while文は前判定ループです
- for文の「更新の式」は繰り返しの度に実行されます
- 後判定ループは1度も繰り返し処理部が実行されないことがあります
- 前判定ループの繰り返し回数は常に偶数回です
- break文を用いると実行中の繰り返し文から抜けます
- 配列は同じ型の複数の変数を管理できます
- 配列の配列要素は条件演算子を用いて確保します
- 1次元配列の最初の配列要素の指定は添え字に1を書きます
- 1次元配列の添え字に-1を指定すると最後の配列要素を参照できます
- 1次元配列はその長さを超えた配列要素には値を代入できません
- 一般にn次元配列の配列要素を指定するにはn個の添え字が必要です
- 配列変数は基本型変数の1つです
- 基本型変数にはint型やdouble型があります
- クラスのメンバの数に制限はありません
- クラス型変数の配列は作ることはできません

■解答欄 1

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20

【問2】 次の for 文の構文とフローチャートです。各式や条件、文が対応する位置をフローチャートから記号で選びなさい。 [各2計8]

構文 for(初期化の式 ; 条件 ; 更新の式) 文;



■解答欄 2

1 初期化の式	2 条件	3 更新の式	4 文
---------	------	--------	-----

【問3】 次の繰り返し文を用いたコードです。各コードを実行したときの画面出力を答えなさい。何も画面出力されない場合は「×」を答えなさい。 [各2計12]

```
(1) int i=0; while(i<2){i++;System.out.print('*');}
```

```
(2) int i=2; while(i<2){i++;System.out.print('*');}
(3) int i=0; while(i!=1){i++;System.out.print('*');}
(4) for(int i=1; i<4; i+=2) System.out.print('*');
(5) for(int i=0,j=0; i-j<3; i+=2,j++) System.out.print('*');
(6) for(int i=0; i<2; i++){
    for(int j=0; j<3; j++) System.out.print('*');
    System.out.print('-');
}
```

■解答欄 3

1	2	3	4	5	6
---	---	---	---	---	---

【問4】 次の配列の初期化を行うコードです。配列を初期化した後に以下の各コードを実行したときの画面出力を答えなさい。 [各2計12]

```
int[] ary1={1,2,3};
int[][] ary2={{0,1},{2},{3,4,5}};
(1) System.out.print(ary1[0]);
(2) System.out.print(ary1[1]);
(3) System.out.print(ary1.length);
(4) System.out.print(ary2[0][1]);
(5) System.out.print(ary2[1][0]);
(6) System.out.print(ary2[ary2.length-1][ary2[ary2.length-1].length-1]);
```

■解答欄 4

1	2	3	4	5	6
---	---	---	---	---	---

【問5】 次のコード1の動作に関する記述です。正しいものには「○」を間違っているものには「×」を答えなさい。 [各2計8]

■コード 1

```
1: import java.io.*;
2: class Final5{
3:     public static void main(String[] args) throws IOException{
4:         BufferedReader br;
5:         br=new BufferedReader(new InputStreamReader(System.in));
6:         int m=-1;
7:         do{
8:             m=Integer.parseInt(br.readLine());
9:             if(m==9)break;
10:        }while(m<1 || m>4);
11:        switch(m){
12:            case 9: System.out.println("PWOFF");break;
13:            default: System.out.println("ACCEPT");break;
14:        }
15:    }
16: }
```

■コード1に関する記述

- キーボードから-1を入力すると何も表示されずに処理が終了します
- キーボードから1を入力するとACCEPTと表示され処理が終了します
- キーボードから5を入力すると再度キーボード入力になります
- キーボードから9を入力するとPWOFFと表示され処理が終了します

■解答欄 5

1	2	3	4
---	---	---	---

【問6】 次のクラスの利用手順に関する説明文です。選択肢より適切な単語を選び空欄を埋めて説明文を完成させなさい。 [各2計8]

【クラスの利用手順について】

クラスを利用する場合、まず [1] を宣言します。 [1] は [2]

を扱う(代入する)ための変数です。 [2] はクラスのメンバー(フィールドなど)を格納するためのメモリ上の領域です。次に [2] を [3] 演算子を用いて確保します。各フィールドを参照する場合には宣言されたクラス型変数名とフィールド名の間 [4] をつけて参照します。

■選択肢

配列変数 オブジェクト 配列要素 添え字 クラス型変数 参照型変数
 ハイフン(-) ピリオド(.) カンマ(,) コロン(:) 四則 new 条件

■解答欄 6

1	2	3	4
---	---	---	---

【問7】 次の 3000g で生まれた乳児の成長日数を計算するコードです。乳児は1日後に30g増えて3030g、2日後に31g増えて3061g、3日後に32g増えて3093gになり、増える量が30g 31g 32g...のように1gずつ増えていきます。キーボードから目標体重(g)を入力すると何日後にその体重以上に成長するか表示します。空欄を適切に埋めてコードを完成させなさい。 [12]

※解答欄には重複があります ※解答欄 [4] は演算子です

※成長体重の変化は簡略しております



■コード 2

```
1: import java.io.*;
2: class Final7{
3:     public static void main(String[] args) throws IOException{
4:         BufferedReader br;
5:         br=new BufferedReader(new InputStreamReader(System.in));
6:         System.out.println("★目標体重(g)を入力してください★");
7:         int target=Integer.parseInt(br.readLine());
8:         int weight=3000;
9:         int [ 1 ] =30;
10:        int days=[ 2 ];
11:        while([ 3 ] [ 4 ] target){
12:            weight+= [ 5 ];
13:            adds++;
14:            [ 6 ] ++;
15:        }
16:        System.out.println(days+"日後に"+weight+"gに成長!!");
17:    }
18: }
```

■実行結果 1

★目標体重(g)を入力してください★

3100

4日後に3126gに成長します

■実行結果 2

★目標体重(g)を入力してください★

4000

25日後に4050gに成長します

■解答欄 7

1	2	3
4	5	6

「お疲れ様でした!!」

☆☆授業評価アンケートのお願い☆☆

授業改善のため、みなさんのご意見を是非お聞かせください。

場所: myFIT(スマホOK) / 期限: 7月20日(水)

★裏面の独自アンケートにもお手数ですが、ご記入お願いします!!