

平成21年度 後其月定其月言式馬庚是真
 科目名 Java プログラミング I
 担当教員 石原真紀夫
 実施日付 1月28日(木) 3時限(1組・2組)
 持ち込み 許可・禁止
 情報工学科 年組学籍番号

氏名

問1 次の各文を読み、文の内容が正しい場合は「○」を間違いの場合は「×」を解答欄に答えなさい。【各2計40】

- (1) 論理演算子&&は「かつ」の演算を行います
- (2) 論理演算子は演算結果として int 型の値を返します
- (3) 論理否定 ! は単項演算子です
- (4) switch 文は分岐文の1つです
- (5) for 文は前判定ループです
- (6) do while 文は後判定ループです
- (7) 初期化の式、条件、更新の式を構文に含むのは while 文です
- (8) 後判定ループは最初に条件を評価してから繰り返し処理部を実行します
- (9) 前判定ループでは1度も繰り返し処理部が実行されないことがあります
- (10) break 文を用いると実行中の繰り返し文から抜けることができます
- (11) continue 文は、実行中の繰り返し処理をもう一度やり直します
- (12) 配列は異なる型の複数の変数を管理します
- (13) 1次元配列の最初の配列要素を指定するには添え字に1を指定します
- (14) 配列の添え字に-1などの負の整数を指定することができます
- (15) 1次元配列はその配列の長さを超えた配列要素に値を代入できます
- (16) 2次元配列の配列要素を指定するには2つの添え字が必要です
- (17) 配列変数は参照型変数の1つです
- (18) 基本型変数には int 型や double 型があります
- (19) クラスのメンバの参照は.(ドット)を用います
- (20) クラス型変数は参照型変数の1つです

問2 次は論理演算子を用いた式を計算するコードです。このコードを実行した直後に変数 b0 から b7 に入っている値(true または false)を解答欄に答えなさい。演算子の優先順位表を裏面に載せています。【各1計8】

```
1: boolean b0, b1, b2, b3, b4, b5, b6, b7;
2: boolean a=true, b=false, c=false;
3: b0 = !!a;
4: b1 = a && b;
```

```
5: b2 = a || c;
6: b3 = !a || b;
7: b4 = a || !a;
8: b5 = b && !b;
9: b6 = !(a && b || c);
10: b7 = a && (b || c);
```

問3 次は1次元配列に代入されている整数を for 文を用いて出力するコードです。下の画面出力となるように空欄に入る適切な数値を答えなさい。【1～4各1】

計4、【5～10各2計12】

```
1: int[] ary={0,1,2,3,4,5,6,7,8,9};
2: int i;
3: for(i= 1;i<10;i+= 2)
4:     System.out.print(ary[i]+ " ");
5: System.out.println("");
6: for(i= 3;i<10;i+= 4)
7:     System.out.print(ary[i]+ " ");
8: System.out.println("");
9: for(i= 5;i< 6;i+= 7)
10: System.out.print(ary[i+1]+ " ");
11: System.out.println("");
12: for(i= 8;i< 9;i+= 10)
13: System.out.print(ary[i]+ " ");
14: System.out.println("");
```

《画面出力》

0 2 4 6 8

1 3 5 7 9

0 1 2

6 7 8

問4 次は不規則な2次元配列の初期化です。下記の設問に答えなさい。【各1計15】

```
int[][] ary={{1, 4, 6, 7}, {2}, {3, 5}};
```

設問1 解答欄に数値を入れてこの配列を図示しなさい。配列要素のない場所には「×」を記入しなさい。

設問2 配列変数 ary に.length 修飾子を次のように指定した場合の画面出力を答えなさい。

- (1) System.out.print(ary.length);
- (2) System.out.print(ary[0].length);
- (3) System.out.print(ary[1].length);

問5 次は switch 文の機能を確認するコードです。変数 i を 0～2 で初期化した場合の画面出力をそれぞれ答えなさい。【各1計9】

(コード A)

```
1: int i= 初期化の値;
2: switch(i){
3:     case 0:System.out.print("a");break;
4:     case 1:System.out.print("a");break;
5:     default:System.out.print("b");
6: }
```

(コード B)

```
1: int i= 初期化の値;
2: switch(i){
3:     case 0:System.out.print("a");break;
4:     case 1:System.out.print("a");
5:     default:System.out.print("b");break;
6: }
```

(コード C)

```
1: int i= 初期化の値;
2: switch(i){
3:     case 0:System.out.print("a");
4:     case 1:System.out.print("a");break;
5:     default:System.out.print("b");break;
6: }
```

問6 次に4パターンの画面出力を示す。下の4つのコードのそれぞれがどのパターンを出力するのか解答欄に対応するパターン番号を答えなさい。【各3計12】

パターン1	パターン2	パターン3	パターン4
0123		123	0
0123	0	023	1
0123	01	013	2
0123	012	012	3

(コード D)

```
1: for(int i=0;i<4;i++){
2:     for(int j=0;j<4;j++){
3:         if(i==j)break;
4:         System.out.print(j);
5:     }
6:     System.out.println();
7: }
```

(コード E)

```
1: for(int i=0;i<4;i++){
2:     for(int j=0;j<4;j++){
3:         System.out.print(j);
4:     }
5:     System.out.println();
6: }
```

(コード F)

```
1: for(int i=0;i<4;i++){
2:     for(int j=0;j<4;j++){
3:         if(i!=j)continue;
4:         System.out.print(j);
5:     }
6: }
```

```
5: }
6: System.out.println();
7: }
(コード G)
1: for(int i=0;i<4;i++){
2:     for(int j=0;j<4;j++){
3:         if(i==j)continue;
4:         System.out.print(j);
5:     }
6: System.out.println();
7: }
```

【解答欄】

問1

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

問2

b0	b1	b2	b3
b4	b5	b6	b7

問3

1	2	3	4	5
6	7	8	9	10

問4 設問1

解答欄	0列目	1列目	2列目	3列目
0行目				
1行目				
2行目				

問4 設問2

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

問5

初期化の値	0	1	2
(コード A)			
(コード B)			
(コード C)			

問6

1 (コード D)	2 (コード E)	3 (コード F)	4 (コード G)
-----------	-----------	-----------	-----------

お疲れ様です。

※試験監督の先生方へ
配布の際にこの面は生徒に見えても良いです

主な演算子の優先順位

演算子	名前	結合規則
<code>++</code>	後置インクリメント	左
<code>--</code>	後置デクリメント	左
<code>!</code>	論理否定	右
<code>~</code>	1 の補数 (反転)	右
<code>+</code>	プラス	右
<code>-</code>	マイナス	右
<code>++</code>	前置インクリメント	右
<code>--</code>	前置デクリメント	右
<code>()</code>	キャスト	右
<code>*</code>	乗算	左
<code>/</code>	除算	左
<code>%</code>	剰余	左
<code>+</code>	加算 (文字列連結)	左
<code>-</code>	減算	左
<code><<</code>	左シフト	左
<code>>></code>	右シフト	左
<code>>>></code>	符号なし右シフト	左
<code>></code>	より大きい	左
<code>>=</code>	以上	左
<code><</code>	未満	左
<code><=</code>	以下	左
<code>==</code>	等価	左
<code>!=</code>	非等価	左
<code>&</code>	ピット論理積	左
<code>^</code>	ピット排他的論理和	左
<code> </code>	ピット論理和	左
<code>&&</code>	論理積	左
<code> </code>	論理和	左
<code>?:</code>	条件	右
<code>=</code>	代入	右
<code>+=,-=など</code>	複合代入演算	右

↑ 優先度高い

↑ 同じ優先度

↓ 優先度低い