

氏名 _____ 点/106点

【問1】次はJavaに関する記述です。各記述が正しい場合は○を、誤っている場合は×を答えなさい。【各2点 計50点】

- (1) スーパークラス型の変数にはサブクラスのオブジェクトを代入できます
- (2) Objectクラスのメンバーはすべてのクラスに継承されます
- (3) メソッドのオーバーライドはポリモーフィズムを実現する方法の一つです
- (4) フィールド(変数)にfinal修飾子をつけると定数になります
- (5) 抽象メソッドとは戻り値を持たないメソッドです
- (6) 抽象クラスのオブジェクトは生成できません
- (7) インタフェースはコンストラクタを持ちません
- (8) インタフェースを拡張してサブインタフェースを宣言できます
- (9) コンパイル時にエラーがでなければ、実行時にはエラーは絶対に発生しません
- (10) 発生した例外は例外クラスのオブジェクトにより表現されます
- (11) 例外が発生したら、プログラムは常に強制終了します
- (12) プログラムは例外を発生させるコードを書くことはできません
- (13) ストリームとは例外処理の流れのことです
- (14) 文字ストリームは画像や音声などのバイナリデータを読み書きするストリームです
- (15) ストリームはディスプレイやファイルなど異なる入出力機器を異なる方法で扱うためのしくみです
- (16) プログラムを実行する時に同時に与えるパラメータをコマンドライン引数といいます
- (17) コマンドライン引数はメインメソッドの仮引数で受け取ります
- (18) コマンドライン引数の数の上限は1です
- (19) スレッドは1つの実行箇所をもつ一連の処理の流れです
- (20) 複数のスレッドは常にそれを開始した順番に終わります
- (21) 複数のスレッドが同じ変数に同時に読み書きを行うと値の不整合が起きる場合があります
- (22) 複数のスレッドの処理を互いに排他的に行うことをオーバーライドといいます
- (23) アプレットはWebブラウザ上で動作します
- (24) アプレットではマルチスレッドを扱うことはできません
- (25) マウスのクリックやドラッグなどのイベントを受け取る処理をイベント処理といいます

【問2】次はJavaで用いる用語です。各用語に関連する選択肢を1つずつ選び記号で答えなさい。【各2点 計12点】

- (1) 例外の送出
- (2) キーボード
- (3) スレッド
- (4) 同期
- (5) アプレット
- (6) インタフェース

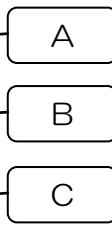
《選択肢》

- (A) throw
- (B) synchronized
- (C) implements
- (D) System.in
- (E) Runnable
- (F) Applet

【問3】次は例外処理の動作を確認するコードです。以下の設問に答えなさい。

```
1: import java.io.*;
2: class Final3{
3:     public static void main(String[] args) throws IOException{
4:         InputStreamReader isr=new InputStreamReader(System.in);
5:         BufferedReader br=new BufferedReader(isr);
```

```
6:         int i;
7:         int[] mi;
8:         try{
9:             i=Integer.parseInt(br.readLine());
10:            mi=new int[i];
11:            System.out.println("E1");
12:        }
13:        catch(NumberFormatException e){
14:            System.out.println("E2");
15:        }
16:        catch(NegativeArraySizeException e){
17:            System.out.println("E3");
18:        }
19:        finally{
20:            System.out.println("E4");
21:        }
22:        System.out.println("E5");
23:    }
24: }
```



設問1 コードを実行して、以下のように各キーボード入力を行った場合の画面出力を正確に答えなさい。【各2点 計6点】

- (1) 1
- (2) -1
- (3) a

設問2 次を示すtry文の各ブロックの機能として適切な説明を1つずつ選択肢から選び記号で答えなさい。【各2点 計6点】

- (1) tryブロック
- (2) catchブロック
- (3) finallyブロック

《選択肢》

- (A) このブロックは、例外が発生した場合の処理を実行します
- (B) このブロックは、try文の後処理として必ず実行します
- (C) このブロックは、例外の発生を監視します

設問3 次を示す各catchブロックをコードに追加する場合、適切な挿入位置をコード中の選択肢A、B、Cから選びなさい。挿入が可能な位置が複数ある場合はそれらをすべて選びなさい。ここで、2つの例外NumberFormatExceptionとNegativeArraySizeExceptionの間にはスーパークラスとサブクラスの間には関係はありません。【各2点 計6点】

- (1) 例外 NumberFormatException のスーパークラスであると同時に、例外 NegativeArraySizeException のスーパークラスでもある例外クラス型変数を仮引数とする catch ブロック
- (2) 例外 NumberFormatException 一方のみのスーパークラスである例外クラス型変数を仮引数とする catch ブロック
- (3) 例外 NegativeArraySizeException 一方のみのスーパークラスである例外クラス型変数を仮引数とする catch ブロック

【問4】次はコマンドライン引数を用いたコードです。下を示す各計算を行わせ結果を表示させたい場合、どのようにコマンドライン引数に値を渡せばよいのか答えなさい。【各2点 計8点】

```
1: class Final4{
2:     public static void main(String[] args){
3:         int[] d=new int[args.length];
4:         for(int i=0;i<d.length;i++){
5:             d[i]=Integer.parseInt(args[i]);
6:         }
7:         switch(d.length){
8:             case 1: System.out.println("円の面積：");
9:                   System.out.println(Math.PI*d[0]*d[0]);
10:                    break;
11:             case 2: System.out.println("三角形の面積：");
12:                    System.out.println(d[0]*d[1]/2.0);
13:                    break;
14:             case 3: System.out.println("値の平均：");
15:                    System.out.println((d[0]+d[1]+d[2])/3.0);
16:                    break;
17:             case 4: System.out.println("2点の距離：");
18:                    double di=Math.pow(d[2]-d[0],2);
19:                    di+=Math.pow(d[3]-d[1],2);
```

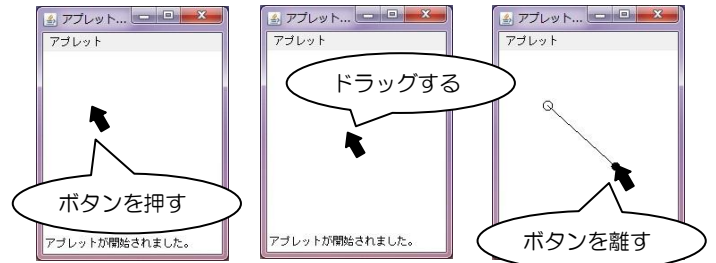
```
20:                    System.out.println(Math.sqrt(di));
21:                    break;
22:                }
23:            }
24:        }
```

《計算》

- (1) 半径1の円の面積
- (2) 底辺3、高さ2の三角形の面積
- (3) 3つの値3,4,5の平均
- (4) 2点(1,1)と(2,2)の間の距離

【問5】次は、マウスのボタンを押した位置を始点、ドラッグしてボタンを離れた位置を終点として、始点に円を終点に塗りつぶした円を描画し、さらに始点と終点の間に線を描画するアプレットです。以下の設問に答えなさい。

《実行画面》



設問1 このアプレットのコードを下に示します。空欄に入る適切な単語を選択肢から1つずつ選びなさい。【各2点 計12点】

《コード》

```
1: import java.applet.Applet;
2: import java.awt.event.MouseListener;
3: import java.awt.event.MouseEvent;
4: import java.awt.Graphics;
5:
6: public class Final5 extends __1__ implements __2__ {
7:     private int mx1,my1,mx2,my2;
8:     public void __3__(){
9:         addMouseListener(this);
10:    }
11:    public void mouseEntered(__4__ e){
12:    }
13:    public void mouseExited(__4__ e){
14:    }
15:    public void mouseClicked(__4__ e){
16:    }
17:    public void mousePressed(__4__ e){
18:        mx1=e.getX();
19:        my1=e.getY();
20:    }
21:    public void mouseReleased(__4__ e){
22:        mx2=e.getX();
23:        my2=e.getY();
24:        repaint();
25:    }
26:    public void __5__(__6__ g){
27:        g.drawOval(mx1-5,my1-5,10,10);
28:        g.fillOval(mx2-5,my2-5,10,10);
29:        g.drawLine(mx1,my1,mx2,my2);
30:    }
31: }
```

《選択肢》

Applet Object Graphics Thread Runnable MouseListener
 MouseEvent System init run start paint join throws

設問2 マウスのイベント処理を行う時、次の各メソッドが呼び出されるタイミングとして適切な選択肢を答えなさい。【各2点 計6点】

- (1) public void mouseEntered(...){...}

- (2) public void mouseExited(...){...}
- (3) public void mouseClicked(...){...}

《選択肢》

- (A) アプレットの画面上でマウスがクリックされたとき
- (B) アプレットの画面上にマウスカーソルが入ったとき
- (C) アプレットの画面上からマウスカーソルが出たとき
- (D) アプレットの画面上でマウスカーソルが一定時間止まったとき
- (E) アプレットの画面上でマウスカーソルが動いたとき

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) ×
(21) ○	(22) ×	(23) ○	(24) ×	(25) ○

【問2】

(1) A	(2) D	(3) E	(4) B	(5) F
(6) C				

【問3 設問1】

(1) E1 E4 E5	(2) E3 E4 E5	(3) E2 E4 E5
--------------------	--------------------	--------------------

【問3 設問2】

(1) C	(2) A	(3) B
-------	-------	-------

【問3 設問3】

(1) C	(2) B, C	(3) C
-------	----------	-------

【問4】

(1) >java Final4 1	(2) >java Final4 3 2
(3) >java Final4 3 4 5	(4) >java Final4 1 1 2 2

【問5 設問1】

(1) Applet	(2) MouseListener	(3) init
(4) MouseEvent	(5) paint	(6) Graphics

【問5 設問2】

(1) B	(2) C	(3) A
-------	-------	-------