

0平成29年度 後期理解度テスト2試験問題 解答  
 科目名 JavaプログラミングⅡ  
 担当教員 石原真紀夫  
 実施日付 1月15日(月) 5時限目(1組2組 D37)  
 持ち込み 許可・禁止  
 情報工学科 年 組 学籍番号  
 氏名

点/102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計50】

- サブクラス型の変数にはスーパークラスのオブジェクトを代入できます
- JavaのすべてのクラスはObjectクラスのメンバーを継承します
- Objectクラス型の変数にはJavaのすべてのクラスのオブジェクトを代入できます
- 抽象メソッドとは処理本体が未定義のメソッドです
- 抽象メソッドは常に戻り値を持ちません
- 抽象クラスのオブジェクトは生成できません
- インタフェースのメンバーは処理内容が定義されたメソッドをもつことができます
- インタフェースを実装してクラスを宣言します
- コンパイル時にエラーがでなければ、プログラムは常に正常終了します
- 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
- 例外が発生したら、プログラムは常に強制終了します
- プログラマは例外を発生(送出)するコードを書くことができます
- ストリームとは異なる入出力機器からのノへのデータを統一的に扱う考え方で
- 文字ストリームは文字ベースのデータを読み書きするストリームです
- 標準入力とはディスプレイのことです
- コマンドライン引数の数は最大2個です
- コマンドライン引数は文字列の配列としてメインメソッドに渡されます
- 1つのスレッドは1つの実行箇所をもつ一連の処理の流れです
- Javaはマルチスレッド処理を記述できません
- マルチスレッド処理では複数のスレッドが並行して実行されます
- 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
- 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発生場合があります
- GUIアプリケーションは線や円などのグラフィックスを描画することはできません
- GUIアプリケーションはボタンやチェックボックスなどのウィンドウ部品を配置できます
- 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計16】

- クラスの拡張の禁止
- キーボード
- 実装
- オブジェクトの属するクラスの検査
- 例外の送出
- ウィンドウとウィンドウ部品を扱うためのライブラリ
- スレッド
- ファイル入力

■選択肢

- final
- Thread
- throw
- AWT
- instanceof
- implements
- FileReader
- System.in

【問3】コマンドライン引数を用いたコード1を以下のように実行した場合の画面出力を正確に答えなさい。【各2計8】

- java Final3
- java Final3 2 1 3
- java Final3 1 2 3 4
- java Final3 0

■コード1

```
1: class Final3{
2:     public static void main(String[] args){
3:         if(args.length==0){
4:             System.out.print("NOデータ");
5:         }else{
6:             int[] num= new int[args.length];
7:             for(int i=0;i<num.length;i++){
8:                 num[i]=Integer.parseInt(args[i]);
9:             }
10:            int sum=0;
11:            for(int i=0;i<num[0];i++){
```

```
12:                sum+=num[i+1];
13:            }
14:            System.out.print(sum);
15:        }
16:    }
17:}
```

【問4】メソッドのオーバーライドに関する下の各コードを実行した場合の画面出力を正確に答えなさい。クラスDeviceとComputer、SmartPhoneの宣言はコード2に示す。【各2計8】

- Device d=new Device("SmartRec");  
System.out.println(d.get());
- Computer c=new Computer("DELXPS",3.5);  
System.out.println(c.get());
- Device d=new SmartPhone("iPhoneX",2.4,"090-1234-5678");  
System.out.println(d.get());
- Computer c=new SmartPhone("Galaxy",2.45,"090-5678-1234");  
System.out.println(c.get());

■コード2

```
1: class Device{
2:     private String name;
3:     public Device(String n){name=n;}
4:     public String get(){
5:         return name;
6:     }
7: }
8: class Computer extends Device{
9:     private double processor;
10:    public Computer(String n, double p){
11:        super(n);
12:        processor=p;
13:    }
14:    public String get(){
15:        return super.get()+"/"+processor+"GHz";
16:    }
17: }
18: class SmartPhone extends Computer{
19:     private String tel;
20:     public SmartPhone(String n, double p, String t){
21:         super(n,p);
22:         tel=t;
23:     }
24:     public String get(){
25:         return super.get()+"/"+tel;
26:     }
27: }
```

【問5】入出力に関する次の説明文の空欄を選択肢から選びなさい。【各2計10】

〔入出力に関する説明文〕

データの入出力は、( 1 ) という概念を用います。( 1 ) は異なる機器からのデータを統一的に扱う考え方で、文字ストリームとバイトストリームがあります。バイトストリームは、画像データや音声データなどの( 2 ) をありのままに読み書きすることができます。( 3 ) とは、ユーザが実行時に入力する文字列を受け取ってプログラムの処理に利用する機能です。与えられた文字列は( 4 ) メソッドの引数として渡されます。たとえば、プログラムSampleを実行する場合、  
 ■java Sample data.txt  
 とすると、( 5 ) という文字列がプログラムに渡されます。

■選択肢

スーパー サブ 抽象 ストリーム 配列 コンストラクタ ポリモーフィズム  
 テキストデータ バイナリデータ コマンドライン引数 Thread Object Graphics  
 Runnable run() paint() main() java Sample data.txt

【問6】GUIアプリケーションに関する次のコード3はマウスカーソルの位置に画(kite.png、縦100ピクセル×横100ピクセル)の画像を追従して表示します。コードの空欄を選択肢から1つずつ選び、埋めてコードを完成させなさい。【各2計10】

■コード3

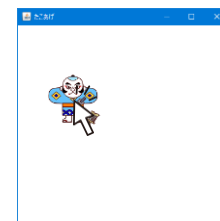
```
1: import java.awt.*;
2: import java.awt.event.*;
3: class Final6 extends Frame implements 1 {
4:     private Image kite;
5:     private int kx,ky;
6:
7:     public Final6(){
8:         super("たこあげ");
9:         Toolkit tk = getToolkit();
10:        kite = tk.getImage("kite.png");
11:        kx=ky=0;
12:        addMouseMotionListener(this);
```

```
13:        addWindowListener(new SimpleWindowAdapter());
14:        setSize(400, 400);
15:        setVisible(true);
16:    }
17:    public void mouseDragged(2 e){}
18:    public void mouseMoved(MouseEvent e){
19:        kx = e.getX();
20:        ky = e.getY();
21:        3 ();
22:    }
23:    public void paint(4 g){
24:        g.drawImage(kite,kx-50,ky-50,this);
25:    }
26:    public static void main(String[] args){
27:        Final6 obj = new Final6();
28:    }
29:    private class 5 extends WindowAdapter{
30:        public void windowClosing(WindowEvent e){
31:            System.exit(0);
32:        }
33:    }
34: }
```

■選択肢

■実行例

Graphics MouseEvent WindowEvent  
 MouseMotionListener Object interface  
 paint repaint Frame final  
 class abstract SimpleWindowAdapter



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) H	(3) F	(4) E	(5) C	(6) D	(7) B	(8) G
-------	-------	-------	-------	-------	-------	-------	-------

【問3】

(1) NOデータ	(2) 4	(3) 2	(4) 0
-----------	-------	-------	-------

【問4】

(1) SmartRec	(2) DELXPS/3.5GHz
(3) iPhoneX/2.4GHz/090-1234-5678	(4) Galaxy/2.45GHz/090-5678-1234

【問5】

(1) ストリーム	(2) バイナリデータ	(3) コマンドライン引数
(4) main()	(5) data.txt	

【問6】

(1) MouseMotionListener	(2) MouseEvent	(3) repaint
(4) Graphics	(5) SimpleWindowAdapter	

お疲れ様でした!!