

平成20年度	<b>後期定期試験問題</b>		
科目名	J a v a プログラミングⅡ		
担当教員	石原真紀夫		
実施日付	1月 23日(金) 3限目(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 次は J a v a に関する記述である。各記述が正しい場合は○を、誤っている場合は×を解答欄に記述せよ。【各2点 計44点】

- 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る
- 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1つである
- メソッドに **final** 修飾子をつけるとそのメソッドはサブクラスに継承されない
- Java では2つ以上のスーパークラスを同時に継承してサブクラスを宣言できる
- 抽象クラスのオブジェクトを生成することはできる
- 抽象メソッドをメンバにもつクラスは抽象クラスとなる
- インタフェースはクラスに実装して使用する
- インタフェースは多重継承を実現する方法の1つである
- インタフェースは継承によりサブインタフェースを作ることができる
- 例外とはコンパイル中に発生する文法の間違いなどのエラーのことである
- 例外は、例外を表すクラスのオブジェクトにより表現される
- 例外が発生したら、本来の処理は直ちに中断される
- 例外が発生した場合、例外に対する適切な処理が行われ、プログラムは常に強制終了する
- ストリームは例外処理の流れのことである
- プログラムを実行する時に同時に与えることができるパラメータをコマンドライン引数という
- コマンドライン引数の数は0個または1個のどちらかでないといけない
- スレッドは1つの実行箇所をもつ一連の処理の流れである
- Java はマルチスレッド処理を記述できない
- 複数のスレッドは常にそれらを開始した順番に終了する
- 同期とは、複数のスレッドの処理を互いに排他的に行うことである
- アプレットは Web ブラウザ上で動作する Java プログラムである
- アプレットはマウスイベントを処理できる

解答欄：

(1) ○	(2) ○	(3) ×	(4) ×
(5) ×	(6) ○	(7) ○	(8) ○
(9) ○	(10) ×	(11) ○	(12) ○

(13) ×	(14) ×	(15) ○	(16) ×
(17) ○	(18) ×	(19) ×	(20) ○
(21) ○	(22) ○		

問2 次は J a v a がもつ代表的な概念または機能である。それぞれの概念・機能に対応する宣言・箇所を下のコード中の記号から1つずつ選び記述せよ。【各2点 計16点】

- オーバーライド
- 定数
- 抽象メソッド
- インタフェースの宣言
- インタフェースの実装
- 例外処理
- 例外の送出
- 同期

解答欄：

(1) f	(2) d	(3) b	(4) a
(5) c	(6) h	(7) g	(8) e

```

class DataEr extends Throwable{
}

interface MathIO{
void setVal(int[] ary) throws DataEr;
void prtVal0;
}

class Div implements MathIO{
private final int Num=2;
private int[] ary;

synchronized public void setVal(int[] p) throws DataEr{
if(p.length!=Num){
DataEr de=new DataEr0;
throw de;
}
ary=p;
}

synchronized public void prtVal0{
int d, m;
try{
d=ary[0]/ary[1];
m=ary[0]-d*ary[1];
System.out.println(ary[0]+"-"+ary[1]+"="+d+"..."+"m);
}catch(ArithmeticException e){
System.out.println("分母が0です");
}catch(NullPointerException e){
System.out.println("データが空です");
}
}
}

```

a → `void setVal(int[] ary) throws DataEr;`  
b → `void prtVal0;`  
c → `private final int Num=2;`  
d → `private int[] ary;`  
e → `synchronized public void setVal(int[] p) throws DataEr{`  
f → `synchronized public void prtVal0{`  
g → `throw de;`  
h → `try{`

例外： <b>ArithmeticException</b> 算術計算で例外的条件(整数をゼロで除算するなど)が発生した場合	例外： <b>NullPointerException</b> オブジェクトを参照しない (null が代入されている) 参照型変数を使おうとした場合
--	---

問3 次のコード中の括弧内で下に示す11通りの各コードを実行した場合の実行画面を正確に答えよ。クラス A、B、C、D、E は解答欄の下に示す。【各3点 計33点】

```

class FinalExam3{
public static void main(String[] args){
}
}

```

(ここで以下の各コードを実行する)

- A a=new A0; a.func10;
- A a=new B0; a.func10;
- B b=new B0; b.func10;
- B b=new B0; b.func20;
- D d=new D0; d.func1(3);
- D d=new D0; d.func2(3);
- D d=new D0; d.func3(3);
- D d=new D0; d.func4(3);
- D d=new D0; d.func5(3);
- E e=new E0; e.func1(5);
- E e=new E0; e.func1(-2);

解答欄：

(1) A1	(2) B	(3) B	(4) A2
(5) 3	(6) 2	(7) 0	(8) 1
(9) 1	(10) E1 E2 E4	(11) E1 E3 E4	

```

class A{
public void func10{System.out.println("A1");}
public void func20{System.out.println("A2");}
}

```

```

class B extends A{
public void func10{System.out.println("B");}
}

class C{
public int x=0, y=1;
}

class D extends C{
public int x=2;
public void func1(int x){System.out.println(x);}
public void func2(int x){System.out.println(this.x);}
public void func3(int x){System.out.println(super.x);}
public void func4(int x){System.out.println(this.y);}
public void func5(int x){System.out.println(super.y);}
}

class E{
public void func1(int n){
try{
System.out.println("E1");
int[] ary=new int[n];
System.out.println("E2");
}catch(NegativeArraySizeException e){
System.out.println("E3");
}finally{
System.out.println("E4");
}
}
}

```

例外：  
**NegativeArraySizeException**  
 負のサイズを持った配列をアプリケーションが作成しようとした場合に送出される

問4 クラス Div (問2を参照) は、整数÷整数の商と余りを計算して画面に出力する機能をもつ。このクラスを用いて8÷3の商と余りを画面に出力するコードを作成しなさい。コードは以下に示す main0メソッドの中身のみ答えよ。【7点】

```

class FinalExam4{
public static void main(String[] args){
}
}

```

(下の解答欄へ記述してください)

解答欄：

```

Div d=new Div0;
int[] data={8, 3};
try{
d.setVal(data);
d.prtVal0;
}catch(DataEr e){
System.out.println("データの形式が正しくありません");
}

```

※例外処理 try 文を記述していない場合、コンパイルエラーとなる。ここでは例外処理を記述してなくても正解とする。

お疲れ様でした。