

氏名

点/104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×を解答欄に記入しなさい。【各2計50】

- (1) クラスはメソッドを複数もつことができない
- (2) メソッドの戻り値がないときは宣言にnullを指定する
- (3) メソッドの戻り値はreturn文を用いて返す
- (4) メソッドの戻り値がないときはvoidを指定する
- (5) メソッドから別のメソッドの実行はできない
- (6) メソッドの戻り値に参照型変数を指定できる
- (7) クラスのpublicメンバは、そのクラスからはアクセスできないメンバである
- (8) クラスのメソッドのみがpublicメンバになることができる
- (9) メソッドのオーバーロードはカプセル化を実現する
- (10) コンストラクタの名前はクラス名と同じである
- (11) コンストラクタのオーバーロードはできない
- (12) コンストラクタからメソッドを呼び出すことができる
- (13) コンストラクタ同士は互いに呼出すことができる
- (14) クラス変数はオブジェクト単位で準備される変数である
- (15) インスタンスメソッド内でクラス変数をアクセスしてもよい
- (16) Doubleクラスはラップクラスである
- (17) Mathクラスはラップクラスである
- (18) クラス型の変数は参照型である
- (19) int型は基本型である
- (20) 参照型変数にはnullを代入できる
- (21) 既存のクラスを拡張して新しいクラスを宣言することができる
- (22) 拡張されたクラスが既存のクラスのメンバを受け継ぐことを継承という
- (23) クラスの拡張における既存のクラスをスーパークラスという
- (24) スーパークラスのprotectedメンバにサブクラスからのアクセスは可能である
- (25) サブクラスはコンストラクタを持つことはできない

【問2】次はJavaで用いる用語や命令の説明です。コード1の記号と合せて、正しい場合は○を、誤っている場合は×を解答欄に記入しなさい。【各2計26】

- (1) AはクラスGadgetを宣言しています
- (2) BはクラスString型変数nameを宣言しています
- (3) Cはコンストラクタの宣言です
- (4) Dは自身のオブジェクトへの参照です
- (5) Eはメソッドの戻り値の型を宣言します
- (6) FはクラスGadgetの戻り値n個のコンストラクタを実行します

- (7) Gは同じクラスからのみアクセスが可能となるアクセス制限を宣言します
- (8) Hはサブクラスです
- (9) Iはスーパークラスのコンストラクタを呼び出します
- (10) Jはメソッドの戻り値がないことを表します
- (11) Kはサブクラスから継承されたメソッドです
- (12) Lはクラス変数やクラスメソッドを宣言するキーワードです
- (13) MはクラスPhoneのオブジェクトを作成します

■コード1

```

class Gadget{
  private String name;
  private int price;

  public Gadget(String n){
    this.name=n;
    this.price=0;
  }
  public Gadget(String n, int p){
    this(n);
    price=p;
  }
  public void show(){
    System.out.print(name+"/"+price+"万円/");
  }
}

class Phone extends Gadget{
  private String tell;

  public Phone(String n, int p, String t){
    super(n, p);
    tell=t;
  }
  public void print(){
    show();
    System.out.print(tell);
  }
}

class Intermediate2{
  public static void main(String[] args){
    Phone p=new Phone("iPhone6s",9,"090-1234-5678");
    p.print();
  }
}
  
```

【問3】次はメソッドのオーバーロードを行うコードです。コンパイルが成功する場合は○を、エラーになる場合は×を解答欄に記入しなさい。【各2計8】

- (1) void show(){...} int show(){...}
- (2) int show(int i){...} int show(int i, int j){...}
- (3) int show(){...} int show(int i){...}
- (4) void show(int a){...} int show(int b){...}

【問4】次に示すStringクラスとMathクラスを用いたコードについて(1)~(5)の各命令を実行したときの画面出力を答えなさい。【各2計10】

```

String str=new String("Computer");
(1) System.out.print(str.length());
(2) System.out.print(str.indexOf('a'));
(3) System.out.print(Math.abs(2));
(4) System.out.print(Math.max(2,-4));
(5) System.out.print(Math.pow(1,2));
  
```

【問5】次の各コードを実行したときの画面出力を正確に解答欄に答えなさい。クラスCardとクラスSugocaの宣言はコード2に示します。【各2計10】

```

class Card;
class Sugoca;
(1) c=new Card();
    c.showC();
(2) c=new Card(2);
    c.showC();
(3) s=new Sugoca();
    s.showC();
    s.showS();
(4) s=new Sugoca(0,1000);
    s.showC();
    s.showS();
(5) s=new Sugoca(4,-1);
    s.showC();
    s.showS();
  
```

■コード2

```

class Card{
  private int id;

  public Card(){
    id=-1;
  }
  public Card(int i){
    this();
    if(i>=1)id=i;
  }
  public void showC(){
    System.out.println("id="+id);
  }
}

class Sugoca extends Card{
  private int charge;

  public Sugoca(){
    charge=0;
  }
  public Sugoca(int i){
    super(i);
    charge=0;
  }
  public Sugoca(int i,int c){
    this(i);
    if(c>=0)charge=c;
  }
  public void showS(){
    System.out.println("charge="+charge);
  }
}
  
```

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)	(2)	(3)	(4)	(5)
(6)	(7)	(8)	(9)	(10)
(11)	(12)	(13)		

【問3】

(1)	(2)	(3)	(4)
-----	-----	-----	-----

【問4】

(1)	(2)	(3)	(4)	(5)
-----	-----	-----	-----	-----

【問5】

(1)	(2)	(3)
id=	id=	id=
		charge=
(4)	(5)	
id=	id=	
charge=	charge=	

お疲れ様でした!!