

計算機ネットワーク工学 I・テスト問題用紙

('04年 7月 29日・13:00 ~ 14:30)

解答上、その他の注意事項

- I. 問題は、問 I~V までである。
- II. 解答用紙の右上の欄に学籍番号・名前を記入すること。
- III. 解答欄を間違えないよう注意すること。
- IV. 解答欄がマス目になっている場合は、1字に1マスを用いること。特に空白にも必ず1マスを用いること
- V. 解答中の文字 (特に a と d) がはっきりと区別できるよう注意すること。
- VI. ノート・プリント・参考書などは持ち込み可である。プリントは冊子としてまとめること。
- VII. 合格は 100 点満点中 60 点以上とする。
(配点 — 期末テスト 80 点, レポート各 20 点)

I. 次の各選択問題に答えよ。(解答の選択肢は必ずしもひとつとは限らない。)

- (i) 次のうち、Java のソースファイルの名前としてふさわしくないのは、どれか？
(A) Hello-1.java (B) Foo1.java (C) Hello1.jav (D) h.java
- (ii) Java ソースコードのコンパイルのためのコマンドとして正しいものはどれか？
(A) java Foo (B) javac Foo (C) java Foo.java (D) javac Foo.java
(E) java Foo.class (F) javac Foo.class
- (iii) また、Java の実行のためのコマンドとして正しいものはどれか？
(ii) の選択肢の中から選べ。
- (iv) 変数 x の値が 12 の時、次のプログラム行を実行した時の出力として、正しいものはどれか？

```
System.out.println("I am 10" + x + " years old.");
```

下の選択肢の中から選べ。

- (A) I am 10 + 12 years old.
(B) I am 22 years old.
(C) I am 1012 years old.
(D) "I am 10"12" years old."
(E) I am 10"12" years old.
(F) I am 10 + 12 + years old.
- (v) 以下のコードで、下線部 (`new Foo("a")`) で作成されたオブジェクトがガーベッジとなり、ガーベッジコレクションの対象になる (つまりプログラムで使用中の変数から参照されなくなる) のは、どの時点か、次にあげる選択肢から正しいものを選べ。
(A) ㉠の行を実行したあと
(B) ㉡の行を実行したあと
(C) ㉢の行を実行したあと
(D) ㉣の行を実行したあと

ファイル: GCTest.java

```
class Foo { // テスト用のどうでもよいクラス
    String x;
    Foo(String s) {
        x = s;
    }
}

public class GCTest {
    public static void main(String[] args) {
        Foo a = new Foo("a");
        Foo b = new Foo("b");
        b = a; // ㉠
        a = new Foo("c"); // ㉡
        b = a; // ㉢
        a = new Foo("d"); // ㉣
    }
}
```

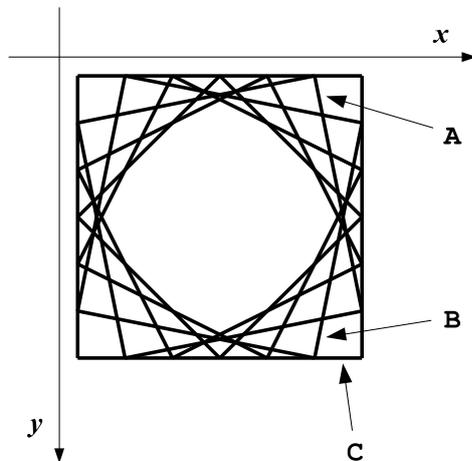
II. 次のプログラムは、画面上にある図形を描くアプレットである。

ファイル: Aste.java

```
import javax.swing.*;
import java.awt.*;

public class Aste extends JApplet {
    public Color[] colors = { Color.red, Color.orange, Color.green,
        Color.cyan, Color.blue, Color.magenta, Color.pink };
    public void paint(Graphics g) {
        int i, j, c=0;
        for (i=0; i<6; i++) {
            int x0=120, y0=-120+40*i;
            for(j=0; j<4; j++) {
                int x1=-y0, y1=x0;
                g.setColor(colors[c%colors.length]);
                g.drawLine(x0+130, y0+130, x1+130, y1+130);
                x0=x1; y0=y1;
                c++;
            }
        }
    }
}
```

右の図中の A, B, C で示される線分の色は何色か？ red, orange, green, cyan, blue, magenta, pink のいずれかで答えよ。(座標軸は、参考のために加えられたもので、実際のプログラムでは描画されない。また、印刷の都合上、上のプログラムで実際に描かれる線より、太く作成している。)



III. つぎに定義されるクラス Animal を継承して、

ファイル: AnimalTest.java (その1)

```
class Animal {
    int age;
    void sound() {
        System.out.print("???", " ");
        age++;
    }
}
```

3つのクラス Dog, Cat, Chicken を作成する。

ファイル: AnimalTest.java (その 2)

```
class Dog  {
    void sound() {
        if (age<2) {
            System.out.print("Kyan, ");
        } else {
            System.out.print("Wan, ");
        }
        age++;
    }
}

class Cat  {
    void sound() {
        System.out.print("Nyaa, " );
        age++;
    }
}

class Chicken  { // 01
    boolean sex; // false -- male, true -- female // 02
    Chicken(boolean s) { sex = s; } // 03
    void sound() { // 04
        if (age<1) { // 05
            System.out.print("Piyo, "); // 06
        } else if (sex) { // 07
            System.out.print("Kokko, "); // 08
        } else { // 09
            System.out.print("Kokekokkooo,"); // 10
        } // 11
        age++; // 12
    } // 13
} // 14
```

ファイル: AnimalTest.java (その 3)

```
public class AnimalTest {
    public static void main(String[] args) {
        Animal[] animals = new Animal[3];
        animals[0] = new Dog();
        animals[1] = new Cat();
        animals[2] = new Chicken(false);

        int i, j;
        for(i=0; i<3; i++) {
            for(j=0; j<3; j++) {
                animals[j].sound();
            }
            System.out.print("¥n");
        }
    }
}
```

- (i) ④の空白（3箇所）を埋めて、クラスの定義を完成させよ。
- (ii) Chicken クラスのフィールド sex の値はコンストラクタでのみ与えることができるものとし、一旦作成した後は、クラスの外からは変更できないようにしたい。Chicken クラスの定義の何行めをどのように変更すれば良いか？（ただし、プログラム中の行の末尾の数字がクラスの中での行数を表す。）
- (iii) このプログラム（AnimalTest クラスの main メソッド）を実行する時、出力はどうか？

IV. 簡易ストップウォッチ、すなわち、次のような動作をするプログラムを Java Applet として作成する。

- 「Start」と「Stop」という二つのボタンがあり、
- プログラムの開始から、約1秒ごとのカウントアップが開始され、
- 「Stop」ボタンを押せば、カウントアップが一時中断する。ただし、0秒に戻らず、再び「Start」ボタンを押した時は、「Stop」した時の値から再びカウントアップする。



次のプログラム中の空欄を埋めて、簡易ストップウォッチ Applet を完成させよ。（なお、この方法では時間は正確に測れないが、それほど精度を必要としないところに使用するので、その点は問題ないと仮定する。）

ファイル: Watch.java

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;

public class Watch extends JApplet
    (i) {
    JButton startBtn, stopBtn;
    Thread thread;
    int count = 0;

    public void init() {
        startBtn = new JButton("Start");
        stopBtn = new JButton("Stop");
        startBtn.addActionListener(this);
        stopBtn.addActionListener(this);
        getContentPane().setLayout(new FlowLayout());
        getContentPane().add(startBtn);
        getContentPane().add(stopBtn);
    }

    public void paint(Graphics g) {
        super.paint(g);
        g.drawString(""+count, 20, 55);
    }

    public void start() {
        if (thread == null) {
            thread = new Thread(this);
            thread.start();
        }
    }

    public void stop() {
        thread = null;
    }

    public void actionPerformed(ActionEvent e) {
        (ii)
    }

    public void run() {
        Thread thisThread = Thread.currentThread();
        while(thread == thisThread) {
            repaint();
            try {
                thisThread.sleep(100); // 100 ミリ秒 = 0.1 秒
            } catch (InterruptedException e) {}
            count++;
        }
    }
}
```

- V. 次のようなクライアント側とサーバ側の2つのプログラムを実行し、通信させることを考える。
(なお、TCP_RO.javaは授業配布プリントの例題7.1.1と同一のプログラムである。)

クライアント側: TCP_RO.java

```
import java.net.*;
import java.io.*;

public class TCP_RO {
    public static void main(String[] argv) {
        try {
            Socket readSocket = new Socket(argv[0], Integer.parseInt(argv[1]));
            InputStream instrm = readSocket.getInputStream();
            while(true) {
                int c = instrm.read();
                if (c==-1) break;
                System.out.write(c);
            }
        } catch (Exception e) {
            e.printStackTrace();
            System.exit(1);
        }
    }
}
```

サーバ側: SimpleServer.java

```
import java.io.*;
import java.net.*;

public class SimpleServer {
    public static void main(String args[]){
        // サーバソケットの準備
        ServerSocket servsock = null;
        Socket sock;
        BufferedReader in;
        // println メソッドが使えるように PrintStream クラスを用いる
        PrintStream out;

        try {
            // サーバ用ソケットの作成
            servsock = new ServerSocket(Integer.parseInt(args[0]));
            while(true){
                sock = servsock.accept(); // 接続要求の受付
                System.out.println("Request from "
                    + (sock.getInetAddress()).getHostName());
                out = new PrintStream(sock.getOutputStream());
                out.println("Hello!");
                out.println("I am using "+args[0]+".");
                sock.close();
            }
        } catch (Exception e){
            e.printStackTrace();
            System.exit(1) ;
        }
    }
}
```

サーバ側のプログラムを IP アドレス: 192.168.0.1,
クライアント側のプログラムを IP アドレス: 192.168.0.2 のマシンで実行するとする。

(i) サーバ側のプログラムを次のようなコマンドで起動するとき、

```
java SimpleServer 8088
```

クライアント側のプログラムは、どのようなコマンドで起動しなければならないか？

(ii) また、この時クライアント側プログラムはどのような出力をするか？

計算機ネットワーク工学 I・テスト解答用紙('04年 7月 29日)

学籍番号		氏名	
------	--	----	--

I. (3点×5)

(i).		(ii).		(iii).	
(iv).		(v).			

II. (5点×3)

A.		B.		C.	
----	--	----	--	----	--

III. (6点, 6点, 8点)

(i).	
(ii).	行目を
	に変えれば良い。
(iii).	

IV. (6点, 10点)

(i).	
(ii).	

V.

(6点, 8点)

(i).	
(ii).	

授業・テストの感想

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....
