

The Midterm Exam

Hsuan-Tien Lin

Department of CSIE, NTU

OOP Class, April 26-27, 2010

Java Night Countdown I

- ① (2%) What is the fully-qualified name of the class `Countdown`?

Ans. `javanight.util.Countdown`

- ② (4%) Complete the method `decrease_minute`.

Ans. You can just copy from `decrease_hour` and modify.

- ③ (4%) Complete the method `get_total_minutes`.

Ans. `return nDay*60*24 + nHour*60 + nMinute;`

- ④ (2%) When compiling the two files above, Java compiler laughs (hahaha) with

```
Timer.java:7: cannot find symbol
```

Adding a line around the place marked with `//` (4) can solve the problem. What is the line?

Ans. `import javanight.util.*;`

- 5 (6%) The hahaha goes away after adding the line in the previous question, but new ones come! What are the minimum changes need to be made in `Countdown.java` to make all the hahaha go away? (*Hint: check access permissions*)

Ans. Declare the methods/constructors called from `Timer` as `public`.

- 6 (12%) Speaking of permissions, what is the tightest permission of each instance variables/methods/constructors of the class `Countdown` to make all the code above work? Note that there are 12 of them.

Ans. `public` for all the methods/constructors in the previous problem, `private` for others.

Java Night Choir I

1 (4%)

```
public Choir(int [] IDs, int nRow){  
    position = new position[1][];  
    position[0] = IDs.clone(); //or simply IDs  
    nStudent = IDs.length;  
    reshape(nRow);  
}
```

2 (4%)

```
public Choir(int [] IDs){  
    this(IDs, 3);  
}
```

Java Night Choir II

8 (4%)

```
public void reshape(int nRow){
    int [][] tmp = new int[nRow][];
    int r, c;
    for(r=0;r<nRow;r++){
        if (r < nStudent/nRow)
            tmp[nRow-r-1] = new int[nStudent/nRow+1];
        else
            tmp[nRow-r-1] = new int[nStudent/nRow];
    }
    int nr = 0, nc = 0;
    for(r=0;r<nRow;r++){
        for(c=0;c<tmp[r].length;c++){
            tmp[r][c] = position[nr][nc]; nc++;
            if (nc >= position[nr].length){
                nr++; nc = 0;
            }
        }
    }
    position = tmp;
}
```

Java Night Choir III

4 (4%)

```
public int getID(int row, int num){
    if (row < position.length && row >= 0
        && num < position[row].length && num >= 0)
        return position[row][num];
    else
        return -1;
}
```

5 (4%)

```
public void showChoir(){
    for(int r=0;r<position.length;r++){
        for(int n=0;n<position[r].length;n++){
            System.out.print(position[r][n] + "_");
            System.out.println();
        }
    }
}
```

- 1 (4%) Fill in the part marked with `/* (1) */` with the tightest access permission modifier.

Ans. `private static void`

- 2 (4%) The method `swap` does not work as expected. How would you change the code from lines 9 to 11 to make the method work?

```
private static void swap(Singer a, Singer b){
    int tmp = a.ID;
    a.ID = b.ID;
    b.ID = tmp;
    tmp = a.score;
    a.score = b.score;
    b.score = tmp;
    /* the wrong code
    Singer tmp = a;
    a = b;
    b = a;
    */
}
```

- 3 (4%) After correcting `swap`, what is the output when the public `compete` is called with an array containing `{(1, 80), (2, 90)}`, where `(1, 90)` means a singer of ID 1 and score 90?

```
1: 2(90) 1(80)
```

- 4 (4%) After correcting `swap`, what is the output when the public `compete` is called with an array containing `{(3, 90), (5, 80), (6, 70), (1, 95)}`?

```
1: 3(90) 5(80) 6(70) 1(95)
2: 3(90) 5(80) 1(95) 6(70)
3: 1(95) 5(80) 3(90) 6(70)
```


- 5 (4%) After correcting `swap`, what is the output when the public `compete` is called with an array containing $\{(3, 90), (5, 80), (1, 85), (2, 60), (4, 70)\}$?

```
1: 3(90) 5(80) 1(85) 2(60) 4(70)
2: 3(90) 5(80) 1(85) 4(70) 2(60)
3: 3(90) 5(80) 1(85) 4(70) 2(60)
4: 3(90) 5(80) 1(85) 4(70) 2(60)
```

Java Night Drinks I

1 (2%) At least how many instance variables does class `BlendedJava` have?

Ans. 3 (milk, caffeine, amount)

2 (3%) There is one place that may result in compile error (hahaha) in the constructor of class `Java`. What is it and how could you fix it by changing only one line?

Ans. `amount` is `private`; can change it to `public` or `protected`.

3 (3%) There is one place that may result in compile error (hahaha) in the constructor of class `BlendedJava`. What is it and how could you fix it?

Ans. The default `BlendedJava()` tries to `super()` while not finding one. Add an empty constructor in `Java`, or add a working constructor in `BlendedJava` with the correct `super` call.

- ④ (4%) We want to modify the action of drinking in class `BlendedJava` as follows. For some `sip` values, you would drink $(ratio * sip)$ from the Java coffee part and $(sip - ratio * sip)$ from the Milk part. Write down such a `drink` method for class `BlendedJava`.

```
public void drink(double sip){
    super.drink(ratio * sip);
    milk.drink(sip - ratio * sip);
}
```

5 (2%)

```
Milk m1 = new Milk();  
Milk m2 = new Milk();  
Java j1 = new Java(3.0, 2.0);  
BlendedJava b1 = new BlendedJava();  
BlendedJava b2 = new BlendedJava();  
b1.milk = m1;  
j1 = b1;  
b1.milk = new FatlessMilk();  
Milk[] marr = new Milk[3];  
marr[0] = b1.milk;  
marr[1] = m2;  
System.out.println(m1 instanceof FatlessMilk);
```

Ans. false

Java Night Drinks IV

6 (2%)

```
Milk m1 = new Milk();
Milk m2 = new Milk();
Java j1 = new Java(3.0, 2.0);
BlendedJava b1 = new BlendedJava();
BlendedJava b2 = new BlendedJava();
b1.milk = m1;
j1 = b1;
b1.milk = new FatlessMilk();
Milk[] marr = new Milk[3];
marr[0] = b1.milk;
marr[1] = m2;
System.out.println(marr instanceof Milk);
```

Ans. hahaha

7 (2%)

```
Milk m1 = new Milk();
Milk m2 = new Milk();
Java j1 = new Java(3.0, 2.0);
BlendedJava b1 = new BlendedJava();
BlendedJava b2 = new BlendedJava();
b1.milk = m1;
j1 = b1;
b1.milk = new FatlessMilk();
Milk[] marr = new Milk[3];
marr[0] = b1.milk;
marr[1] = m2;
System.out.println(marr[1] == marr[0]);
```

Ans. false

Java Night Drinks VI

8 (2%)

```
Milk m1 = new Milk();
Milk m2 = new Milk();
Java j1 = new Java(3.0, 2.0);
BlendedJava b1 = new BlendedJava();
BlendedJava b2 = new BlendedJava();
b1.milk = m1;
j1 = b1;
b1.milk = new FatlessMilk();
Milk[] marr = new Milk[3];
marr[0] = b1.milk;
marr[1] = m2;
System.out.println(marr[2].getAmount());
```

Ans. ohohoh

9 (2%)

```
Milk m1 = new Milk();
Milk m2 = new Milk();
Java j1 = new Java(3.0, 2.0);
BlendedJava b1 = new BlendedJava();
BlendedJava b2 = new BlendedJava();
b1.milk = m1;
j1 = b1;
b1.milk = new FatlessMilk();
Milk[] marr = new Milk[3];
marr[0] = b1.milk;
marr[1] = m2;
System.out.println(b1 instanceof Object);
```

Ans. true

Java Night Drinks VIII

10 (2%)

```
Milk m1 = new Milk();
Milk m2 = new Milk();
Java j1 = new Java(3.0, 2.0);
BlendedJava b1 = new BlendedJava();
BlendedJava b2 = new BlendedJava();
b1.milk = m1;
j1 = b1;
b1.milk = new FatlessMilk();
Milk[] marr = new Milk[3];
marr[0] = b1.milk;
marr[1] = m2;
System.out.println(j1 instanceof BlendedJava);
```

Ans. true

Java Night Drinks IX

11 (2%)

```
Milk m1 = new Milk();
Milk m2 = new Milk();
Java j1 = new Java(3.0, 2.0);
BlendedJava b1 = new BlendedJava();
BlendedJava b2 = new BlendedJava();
b1.milk = m1;
j1 = b1;
b1.milk = new FatlessMilk();
Milk[] marr = new Milk[3];
marr[0] = b1.milk;
marr[1] = m2;
System.out.println(b2.milk);
```

Ans. null

12 (2%)

```
Milk m1 = new Milk();
Milk m2 = new Milk();
Java j1 = new Java(3.0, 2.0);
BlendedJava b1 = new BlendedJava();
BlendedJava b2 = new BlendedJava();
b1.milk = m1;
j1 = b1;
b1.milk = new FatlessMilk();
Milk[] marr = new Milk[3];
marr[0] = b1.milk;
marr[1] = m2;
System.out.println(marr[0] instanceof BlendedJava);
```

Ans. hahaha

13 (2%)

```
Milk m1 = new Milk();
Milk m2 = new Milk();
Java j1 = new Java(3.0, 2.0);
BlendedJava b1 = new BlendedJava();
BlendedJava b2 = new BlendedJava();
b1.milk = m1;
j1 = b1;
b1.milk = new FatlessMilk();
Milk[] marr = new Milk[3];
marr[0] = b1.milk;
marr[1] = m2;
System.out.println(j1.caffeine == b1.caffeine);
```

Ans. true

Brainstorming Time I

Ans. ???