

Comp 110
Midterm
Examination
Answers
19 November 2013

NAME: _____ **ANSWERS** _____

Answer all questions on the test paper. No calculators or any other electronic devices may be used during the test. Show all of your work.

1. Describe the main difference between do-until loops and while loops. (5 pts)

a) do-until loops

enter loop before satisfying the condition

b) while loops

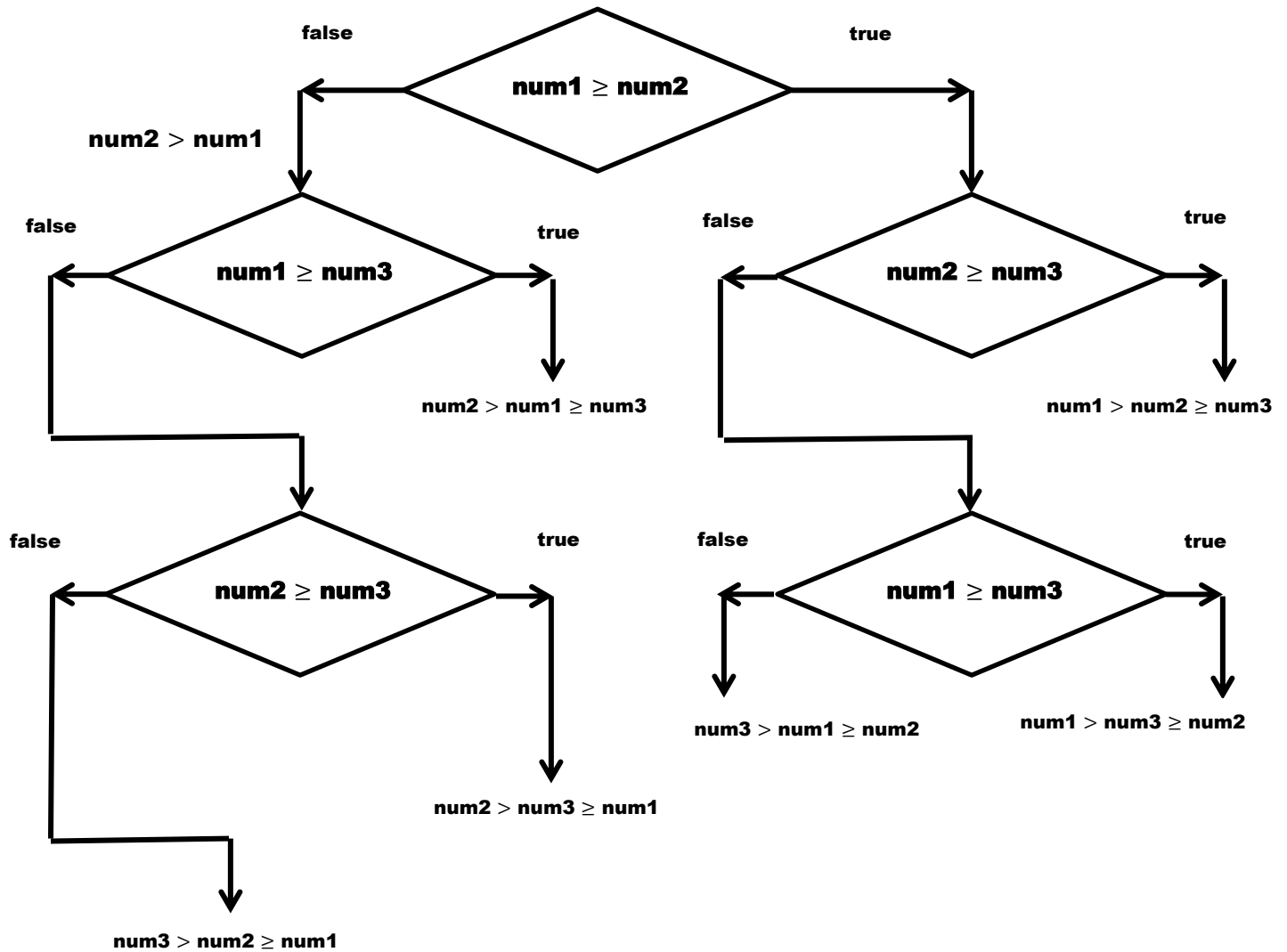
satisfying the condition before entering the loop

2. Using the ternary operator “(...) ? ___ : ___ ;” i.e., the question mark operator, write a single, concise, compound Java statement that sets (5 pts)

- $z = 0$ if $x > 0$
- $z = -2$ if $x == 0$
- $z = 1$ if $x < 0$

- a) $(x > 0)? z = 0 : ((x == 0)? z = -2) : z = 1;$
b) $(x > 0)? z = 0 : ((x < 0)? z = 1) : z = -2;$
c) $(x == 0)? z = -2 : ((x > 0)? z = 0) : z = 1;$
d) $(x == 0)? z = -2 : ((x < 0)? z = 1) : z = 0;$
e) $(x < 0)? z = 1 : ((x > 0)? z = 0) : z = -2;$
f) $(x < 0)? z = 1 : ((x == 0)? z = -2) : z = 0$
g) ;

3. Write a program fragment, i.e., pseudocode, Java code or flow charts, sorting three integers stored in the variables num1, num2, num3 according to value, largest to smallest. (10 pts)



4. Given the program fragment

(5pts)

```
int i = 0, s = 0;
while( i < 17 )
{
    if ( i % 3 == 0 ) continue;
    if ( i % 5 == 0 ) continue;
    s += i;
}
System.out.println( "s : " + s );
```

produce the output line.

Never terminates, thus no output line.

```
int i = 0, s = 0;
while( i < 17 )
{
    if ( i % 3 == 0 ) continue;
    if ( i % 5 == 0 ) continue;
    s += i;
    i++;
}
System.out.println( "s : " + s );
```

Never terminates, thus no output line.

```
int i = 0, s = 0;
while( i++ < 17 )
{
    if ( i % 3 == 0 ) continue;
    if ( i % 5 == 0 ) continue;
    s += i;
}
System.out.println( "s : " + s );
```

Output s: 76

5. Describe the difference between a class and an object.

(5 pts)

a) class – pattern, design, bag,

b) object – instance of a class

6. Given the following definitions and operations provide the resulting values for the indicated items (15 pts)

a. `int n = 6`
`final int m = 13;`
`n /= m;`
`m /= n;`

n m // compiler error

b. `int n = 21;`
`final int m = 3;`
`int p, q;`
`p = n/m;`
`q = n%m;`

p q

c. `int n = 6;`
`int m = 9;`
`int p = m++ - ++n;`

p n m

d. The expression `(char) ('a' + Math.random() * ('p' - 'm' + 1));`
returns a random character between a and d .

e. `int x = 0; int y = (x > 0) ? 10 : 20;` y

7. Write the output of the following program in the box below.

(15 pts)

```
public class Test
{
    public static void main(String[ ] args)
    {
        int x = 1;
        int y = 2;

        System.out.println("Before swap, x is " + x + "and y is " + y );
        swap(x, y);
        System.out.println("After swap, x is " + x + "and y is " + y );
    }

    public static void swap( int i, int j)
    {
        System.out.println("Inside swap method, before swap, i is " + i + " and j is " + j );

        int temp = i;
        i = j;
        j = temp;

        System.out.println("Inside swap method, after swap, i is " + i + " and j is " + j );
    }
}
```

Output:

Before swap, x is 1 and y is 2
Inside swap method, before swap, i is 1 and j is 2
Inside swap method, after swap, i is 2 and j is 1
After swap, x is 1 and y is 2

8. For the given the method, write the header and signature in the space below. (5 pts)

```
public static void increase(int[ ] x)
{
    for (int i = 0; i < x.length; i++) x[i]++;
}
```

Page 179 Method Headers &
Signatures

Header

```
public static void increase(int[ ] x)
```

Signature

```
increase(int[ ] x
```

9. For the given the method, write the header and signature in the space below. (5 pts)

```
public static void increase(int y) {y++; }
```

Page 179 Method Headers &
Signatures

Header

```
public static void increase(int y)
```

Signature

```
increase(int y)
```


10. Given the following definition of a Circle,

```
public class Circle
{
    private double radius;
    private static int numberOfObjects = 0;

    public Circle
    {
        numberOfObjects++;
    }

    public Circle(double radius)
    {
        this.radius = radius;
        numberOfObjects++;
    }

    public double getRadius( ) return radius;

    public void setRadius( double radius)
    {
        this.radius = (radius >= 0) ? radius : 0;
    }

    public static int getNumberOfObjects( )
    {
        return numberOfObjects;
    }

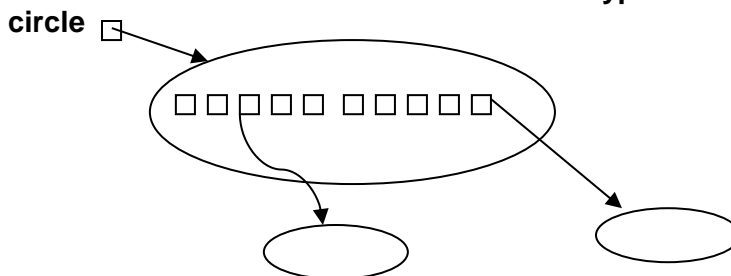
    public double getArea( ) return radius * radius * Math.PI;
}
```

what does the following statement provide?

(5 pts)

```
Circle [ ] circle = new Circle[10];
```

Reference variable of type Circle holding an array object of size ten, i.e., circle,



11. Describe the difference between (5 pts)
- a. the reference variable Integer
and
 - b. the primitive variable int

ANSWER

- a. the reference variable Integer has the potential to hold an object of type int as well as many methods to manipulate the object's contents
- b. the primitive variable int just has the potential to hold an integer value

12. Describe the operational difference between (5 pts)
- a. static variables
and
 - b. instant variables

ANSWER

- a. static variables, i.e., class variables, hold values at the class level; the common value of each static variable is available to all objects as well as to the class
- b. instant variables can only hold values at the object level; each object's value to a variable with the same name may be different

13. Describe the operational difference between (5 pts)
- a. static methods
and
 - b. instant methods

ANSWER

- a. static methods are available for use without creating an object of that class
some classes, e.g., the Math class are intentionally designed to hold static methods
- b. instant methods can only be made available after an object is created

14. Given the following binary search algorithm

(10 pts)

```

public static int binarySearch( int[] list, int key )
{
    int low = 0;
    int high = list.length - 1;

    while( high >= low )
    {
        int mid = ( low + high ) / 2;
        if ( key < list[ mid ] ) high = mid - 1;
        else if ( key == list[ mid ] ) return mid;
        else low = mid + 1;
    }
    return -1; // not found
}
    
```

trace the search algorithm for the keys 69 and 34

Use the following array, written with black numbers, e.g., 59 and with an attached index set, written with red numbers with a yellow background, e.g., 8

| | | | | | | | | | | | | | | | |
|---|---|---|----|----|----|----|----|----|----|----|----|----|----|----|-----|
| 2 | 4 | 7 | 10 | 11 | 28 | 33 | 45 | 59 | 61 | 66 | 69 | 71 | 79 | 91 | 109 |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |

and the following table for tracing the variables low, high and mid from the keys selected below

| key == 69 | | | | key == 34 | | |
|-----------|-----|------|--|-----------|-----|------|
| low | mid | high | | low | mid | high |
| 0 | 7 | 15 | | 0 | 7 | 15 |
| 8 | 11 | 15 | | 0 | 3 | 6 |
| | | | | 4 | 5 | 6 |
| | | | | 6 | 6 | 6 |
| | | | | 7 | | 6 |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |

return 11
list[11] == 69

return -1
// not found