

CS 1323-020: Introduction to Computer Programming
Final Exam
December 9, 2014

General instructions:

- Please wait to open this exam booklet until you are told to do so.
- This examination booklet has 21 pages. You also have been issued a bubble sheet.
- Write your name, university ID number and date, and sign your name below. Also, write your name and ID number on your bubble sheet, and fill in the bubbles for your ID.
- The exam is open book and open notes, but is closed electronic device. The only exception is that you may use an electronic device to display a PDF copy of the book (all communication must be turned off and no other applications may be used). c
- The exam is worth a total of 200 points (and 20% of your final grade). a
- You have 2 hours to complete the exam. Be a smart test taker: if you get stuck on one problem go on to the next. b
- Use your bubble sheet to answer all multiple-choice questions. Make sure that the question number and the bubble row number match when you are answering each question. Use the provided space in this exam booklet to answer the coding questions. c

On my honor, I affirm that I have neither given nor received inappropriate aid in the completion of this exam.

Signature: _____

Name: _____

ID Number: _____

Date: _____

Question	Points	Score
Variables, Types and Conversions	16	
Mathematical Operators	24	
Conditionals and Logic	16	
Methods	20	
Objects and Classes	28	
Loops, Arrays and Search	20	
Nested Loops	12	
Coding	64	
Total:	200	

Part I. Variables, Types and Conversions

1. (4 points) The String class uses what primitive type to represent the contents of the string?

A. int B. boolean C. double **D. char** E. Answer not shown

2. (4 points) Given the following block of code. Which line contains an error?

```
1 int foo = 5;
2 int bar = 42;
3 double foo = 3.5;
4 double baz = 3.1;
```

A. Line 1 B. Line 2 **C. Line 3** D. Line 4 E. There are no errors

3. (4 points) What type of data would you use to represent the current speed that a baby is moving an assistive robot?

A. int B. boolean **C. double** D. String E. Answer not shown

4. (4 points) Given the following block of code. Which line contains an error?

```
1 int foo = 17;
2 double bar = 2.7;
3 bar = foo * 2;
4 foo = bar / 2;
```

A. Line 1 B. Line 2 C. Line 3 **D. Line 4** E. There are no errors

Part II. Mathematical Operators

5. (4 points) What is printed by this block of code?

```
int a = 13;
int b = a + 2 * 3;
int c = b / 4 + 5;

System.out.println(c);
```

- A. 2 B. 5 C. 8 D. 16 E. Answer not shown

6. (4 points) What is printed by this block of code?

```
double a = 18.5;
double b = a / 2;

System.out.println(b);
```

- A. 9 B. 9.0 C. **9.25** D. 9.75 E. Answer not shown

7. (4 points) What is printed by this block of code?

```
int a = 11;
double b = a / 2;
double c = b * 2;

System.out.println(c);
```

- A. 5.5 B. **10.0** C. 11.0 D. 11.5 E. Answer not shown

8. (4 points) What is printed by this block of code?

```
int a = 17;
int b = a - 3;
double c = b / 4.0 + 3;

System.out.println(c);
```

- A. 6.0 B. **6.5** C. 7.0 D. 7.25 E. Answer not shown

9. (4 points) What is printed by this block of code?

```
int j = 17;
int k = 4;

System.out.println(j % k);
```

- A. **1** B. 2 C. 3 D. 4 E. Answer not shown

10. (4 points) What is printed by this block of code?

```
int a = 7;  
int b = a / 2;  
int c = b * 2;  
  
System.out.println(c);
```

A. 3 B. 3.5 C. 6 D. 7 E. Answer not shown

Part III. Conditionals and Logic

11. (4 points) What is printed by this block of code?

```
int p = 49;
int q = 51;
int r = 53;

if(r / 5 == 9)
{
    System.out.println(1);
} else if(p - q > r) {
    System.out.println(2);
} else if(q % 3 == 0){
    System.out.println(3);
} else{
    System.out.println(4);
}
```

- A. 1 B. 2 **C. 3** D. 4 E. Answer not shown

12. (4 points) What is printed by this block of code?

```
double m = 17.3;
double n = 2.71;

if(m > n)
{
    System.out.println(m - 3.3);
} else{
    System.out.println(n + 4.1);
}
```

- A. -1.41 B. 6.81 **C. 14.0** D. 21.4 E. Answer not shown

13. (4 points) What is printed by this block of code?

```
int p = 41;
int q = 43;
int r = 47;

if(q > r && r < p || p < q)
{
    System.out.println(1);
} else if(r == q) {
    System.out.println(2);
} else if(r > q){
    System.out.println(3);
} else{
    System.out.println(4);
}
```

- A. 1** B. 2 C. 3 D. 4 E. Answer not shown

14. (4 points) What is printed by this block of code?

```
int s = 29;
int t = 31;
int u = 37;

if(t > u && s < t)
{
    System.out.println(1);
} else if(s > t) {
    System.out.println(2);
} else if(u < t || s < u){
    System.out.println(3);
} else{
    System.out.println(4);
}
```

A. 1 B. 2 **C. 3** D. 4 E. Answer not shown

Part IV. Methods

15. (4 points) What is printed by this program?

```
public static int bar(int t)
{
    return t % 3;
}

public static void main(String[] args)
{
    System.out.println(bar(bar(bar(19))));
}
```

- A. 0 **B. 1** C. 2 D. 3 E. Answer not shown

16. (4 points) What is printed by this program?

```
public static int baz(int a, int b)
{
    a = a - 3;
    b = a / 2;
    return b;
}

public static void main(String[] args)
{
    int a = 10;
    int b = 17;
    int c = baz(b, a);
    int d = baz(c, b);

    System.out.println(d);
}
```

- A. 0 B. 1 **C. 2** D. 3 E. Answer not shown

17. (4 points) What is printed by this program?

```
public static int foobar(int j, int k)
{
    return j + k;
}

public static void main(String[] args)
{
    int j = 2;
    int k = 3;

    j = foobar(j, k);
    k = foobar(k, j);
    System.out.println(foobar(j,k));
}
```

- A. 0 B. 5 C. 10 **D. 13** E. Answer not shown

18. (4 points) What is printed by this program?

```
public static int m1(int a)
{
    return a * 2;
}

public static int m2(int a)
{
    return m1(a) + a / 2;
}

public static void main(String[] args)
{
    int a = 3;

    a = m2(a);
    a = m2(a);

    System.out.println(a);
}
```

A. 3 B. 7 C. 17 D. 27 E. Answer not shown

19. (4 points) What is printed by this program?

```
public static String stringModifier(String s)
{
    return s.toLowerCase() + s.toUpperCase();
}

public static void main(String[] args)
{
    String strg = "Foo";

    System.out.println(stringModifier(strg) + strg);
}
```

A. Foo B. FOOfoo C. fooFoo D. fooFOOfoo E. Answer not shown

Solution: fooFOOfoo

Part V. Objects and Classes

20. (4 points) What is printed by this program?

```
public static String stringModifier(String s)
{
    s = s.toUpperCase();
    return s;
}

public static void main(String[] args)
{
    String strg1 = "baR";
    String strg2 = stringModifier(strg1);

    System.out.println(strg1);
}
```

- A. **baR** B. BAR C. bar D. Bar E. Answer not shown

Consider the following class definition...

```
1 public class Point
2 {
3     private double x;
4     private double y;
5
6     public Point(double xValue, double yValue)
7     {
8         x = xValue;
9         y = yValue;
10    }
11
12    public double compute(Point p)
13    {
14        return Math.pow(x - p.x, 2) + Math.pow(y - p.y, 2);
15    }
16
17    public Point combine(Point p)
18    {
19        return new Point(p.x + x, p.y + y);
20    }
21
22    public String toString()
23    {
24        return "(" + x + ", " + y + ")";
25    }
26
27    public static void main(String[] args)
28    {
29        Point p1 = new Point(2,4);
30        Point p2 = new Point(4,7);
31
32        System.out.println(p1.compute(p2) + p2.compute(p2));
33
34        System.out.println(p2);
35
36        System.out.println(p2.combine(p1));
37    }
38 }
```

21. (4 points) What is printed by line 32 of this program?
A. 0.0 B. 5.0 **C. 13.0** D. 26.0 E. Answer not shown
22. (4 points) What is printed by line 34 of this program?
A. (2.0, 4.0) **B. (4.0, 7.0)** C. (4.0, 2.0) D. (7.0, 4.0)
E. Answer not shown
23. (4 points) What is printed by line 36 of this program?
A. (2.0, 4.0) B. (4.0, 7.0) C. (11.0, 6.0) **D. (6.0, 11.0)**
E. Answer not shown

24. (4 points) What is printed by this program?

```
public static StringBuilder stringModifier3(StringBuilder s)
{
    s.reverse();
    return s;
}

public static void main(String[] args)
{
    StringBuilder strg1 = new StringBuilder("baR");
    StringBuilder strg2 = stringModifier3(strg1);

    System.out.println(strg1.toString() + strg2.toString());
}
```

A. baR B. Rab C. baRRab **D. RabRab** E. Answer not shown

25. (4 points) What type of variable would be used to store a temporary value used to decide whether one circle is contained within another?

A. local B. class C. final D. instance E. Answer not shown

26. (4 points) What type of variable would be used to store the center of a circle object?

A. local B. class C. final **D. instance** E. Answer not shown

Part VI. Loops, Arrays and Search

Consider the following program ...

```
1  public static int check(Integer [] list)
2  {
3      for(int i = list.length-1; i > 0; i --)
4      {
5          if(list[i] + list[i-1] > 11)
6              return i;
7      }
8      return -1;
9  }
10
11 public static void main(String [] args)
12 {
13     Integer [] list1 = {6, 7, 3, 6, 7, 4, 7};
14     Integer [] list2 = {6, 4, 3, 7, 2, 1, 9};
15
16     System.out.println(check(list1));
17     System.out.println(check(list2));
18 }
```

27. (4 points) What is printed by line 16 of the above program?
A. 1 B. 2 C. 3 **D. 4** E. Answer not shown
28. (4 points) What is printed by line 17 of the above program?
A. 1 B. 2 C. 3 D. 4 **E. Answer not shown**

29. (4 points) What result is printed by this program?

```
public static int foo(int [] things, int a)
{
    int top = 0;
    int bottom = things.length;
    int k = 0;
    while(top != bottom)
    {
        int center = (top+bottom)/2;
        if(a == things[center])
        {
            return k;
        }else if(a > things[center])
        {
            top = center+1;
        }else
        {
            bottom = center;
        }
        ++k;
    }
    return k;
}

public static void main(String [] args)
{
    int a = 18;
    int [] b = {5, 7, 11, 13, 17, 19, 23};

    System.out.println(foo(b, a));
}
```

A. 1 B. 2 C. 3 D. 4 E. Answer not shown

30. (4 points) What is printed by this program?

```
public static void move(String [] list)
{
    String a = list[0];

    for(int i = 0; i < list.length -1; ++i)
    {
        list[i] = list[i+1];
    }
    list[list.length-1] = a;
}

public static void main(String [] args)
{
    String [] list = {"Lando", "Vir", "Cartagia", "Dius"};
    move(list);

    System.out.println(list[0] + "; " + list[3]);
}
```

A. Lando; Vir B. Lando; Dius C. Vir; Cartagia D. Vir; Lando
E. Answer not shown

31. (4 points) What is printed by this block of code?

```
ArrayList<Double> doubles = new ArrayList<Double>();  
  
doubles.add(0, 5.1);  
doubles.add(0, 2.7);  
doubles.add(1, 3.1);  
doubles.add(0, 3.8);  
  
System.out.println(doubles.get(0) + doubles.get(2));
```

A. 6.9 B. 8.2 C. 3.83.1 D. 3.15.1 E. Answer not shown

Part VII. Nested Loops

32. (4 points) What result is printed by this code block?

```
int[] things = {2, 1, 4};
int val = 0;

for(int i = 0; i < things.length - 1; ++i)
{
    for(int j = i; j <= i + 1; ++j)
    {
        val += things[i] * things[j];
    }
}
System.out.println(val);
```

A. 5 B. 7 C. 11 D. 19 E. Answer not shown

33. (4 points) What result is printed by this code block?

```
int[] things = {3, 8, 1, 1};

for(int i = 0; i < length.things; ++i)
{
    for(int j = length.things-1; j >= 0; j -= 1; )
    {
        System.out.print(things[j]);
    }
}
System.out.println("");
```

A. 113811381138 B. 381138113811 C. 1138113811381138
D. 3811381138113811 E. Answer not shown

Solution: Correct answer: 1183118311831183
Also: compiler will not like "length.things"

34. (4 points) What result is printed by this code block?

```
int[] things = {3, 1, 2};
int val = 0;

for(int i = 0; i < things.length; ++i)
{
    for(int j = i - 1; j >= 0; j -= 1)
    {
        val += things[i] * things[j];
    }
}
System.out.println(val);
```

A. 3 B. 11 C. 14 D. 25 E. Answer not shown

Part VIII. Coding

35. (16 points) Write a **method** that takes as input an array of ints and an int, and returns the number of times that the int **does not occur** in the **even** elements of the array (counting from zero).

Examples:

- (2, 3, 7, 2, 9), 2 → 2
- (1, 2, 5, 1), 6 → 2
- (5, 3, 7, 1, 5), 5 → 1

Solution:

```
public static int countNegativeQuery(int[] vals, int query)
{
    int counter = 0;

    for(int i = 0; i < vals.length; i += 2)
    {
        if(vals[i] != query) ++counter;
    }
    return counter;
}
```

Consider the following partial class definition:

```
public class MyRectangle {
    // We assume that the segments of the rectangle are either
    // horizontal or vertical
    private double[] x;
    private double[] y;

    /**
     * x1, y1 is the coordinate of the upper left corner of the rectangle
     * x2, y2 is the coordinate of the lower right corner of the rectangle
     */
    public MyRectangle(double x1, double y1, double x2, double y2)
    {
        x = new double[] {x1, x2};
        y = new double[] {y1, y2};
    }
}
```

36. (16 points) Write the following **class methods** in the space below:

```
/**
 * @return the area of the rectangle
 */
public double area(){

}

/**
 * @return the perimeter of the rectangle
 */
public double perimeter(){

}

}
```

Hints: `double Math.abs(double a)` returns the absolute value of the parameter.

Solution:

```
public double area(){
    return Math.abs((x[0]-x[1]) * (y[0]-y[1]));
}

public double perimeter(){
    return 2.0 * (Math.abs(x[0]-x[1]) + Math.abs(y[0]-y[1]));
}
```

37. (16 points) Assume that you are provided with an implementation of the following method that takes as input two sorted lists of doubles and returns a single list of doubles containing all elements of the two lists:

```
public static double [] merge(double [] list1, double [] list2)
```

Using the above method, write a **new method** that takes as input a single array of doubles and returns the same doubles in a sorted array. You may only use a single for() loop.

Examples:

- (2.4, 3.5, 7.1, 2.0, 9.8) → (2.0, 2.4, 3.5, 7.1, 9.8)
- (1.1, 2.7, 5.8, 1.0) → (1.0, 1.1, 2.7, 5.8)

Solution:

```
public static double [] sort(double [] list){
    double [] newList = new double [0];
    double [] shortList = new double [1];

    for(int i = 0; i < list.length; ++i){
        shortList [0] = list [i];
        newList = merge(newList, shortList);
    }

    return newList;
}
```

38. (16 points) Write a **method** that takes as input an array of ints. For each element of the array, the method prints a line that contains the indicated number of “#” signs. If the value is not positive, then no characters are printed.

Examples:

- `displayHistogram(new int[] {3, 7, 2})` would print out:

```
###  
#####  
##
```

- `displayHistogram(new int[] {6, -1, 3, 0, 4})` would print out:

```
#####  
###  
####
```

Solution:

```
public static void displayHistogram(int[] vals)
{
    for(int i = 0; i < vals.length; ++i)
    {
        for(int j = 0; j < vals[i]; ++j)
        {
            System.out.print("#");
        }
        System.out.println("");
    }
}
```