

CSCE 145 – PRACTICE Midterm Exam 2 – 25 March
2008

You have 50 minutes. Look over the test first. Divide points into time to estimate how much time you ought to spend on any given question.

Justify any answer you give.

You may wish to make simplifying assumptions. **STATE THOSE ASSUMPTIONS CLEARLY.** If the assumptions are entirely reasonable (that is, they turn an otherwise very difficult problem into a problem that is reasonable for an exam) then this is perfectly acceptable. If your assumptions turn an otherwise reasonable problem into a relatively trivial problem, then I will not grant full marks for the answer.

NAME:

1. (15)	5. (7)	9. (10)
2. (10)	6. (8)	10. (5)
3. (10)	7. (10)	
4. (10)	8. (15)	Total

1. (15 points = 5 + 5 + 5) Identify at least one *instance variable* and at least one *temporary variable* in the following code. What properties do these possess that makes them instance or temporary variables?

```
public class myRobot extends Robot
{
    int myZorkmids;
    int companyZorkmids;
    public myRobot(int howManyOfMine, int howManyOfMyCompany)
    {
        this.companyZorkmids = howManyOfMyCompany;
        this.myZorkmids = howManyOfMine;
    }

    public void printZorkmids()
    {
        int myZorkmids = 1357;
        int companyZorkmids = 2468;
        System.out.println("my Zorkmid count is " + myZorkmids);
    }
}
```

2. (10 points) Write a method that produces the following output. You must use at least one loop and must not simply use `println` statements that produce exactly the output. You should use one or more `printf` statements to produce tabular output; the first `int` value should be printed in a field four spaces wide and the second should be printed in a field five spaces wide, with a space in between each field.

```
2   4
3   9
4  16
5  25
6  36
7  49
8  64
9  81
```

3. (10 points) Write a method that produces the following output. You must use at least one loop and must not simply use `println` statements that produce exactly the output. You should use one or more `printf` statements to produce tabular output; the first value should be printed as an `int` in a field four spaces wide and the second should be printed as a `double` in a field eight spaces wide with one place to the right of the decimal point, and with a space in between each field. READ THIS OUTPUT CAREFULLY!!!! YOU ARE PERMITTED TO "HARD CODE" THE TEST TO GET THIS OUTPUT EXACTLY AS WRITTEN.

```
2 6.0
3 9.0
4 12.0
6 18.0
7 21.0
8 24.0
9 27.0
```

4. (10 points) What is the difference between the following two loops?

```
// loop one
int i;
i = 0;
j = 1;
while(i < 10)
{
    j = computeSomething(i);
    System.out.println("i and j are " + i + " and " + j);
}
```

```
// loop two
int i;
i = 0;
j = 1;
while(i < 10)
{
    System.out.println("i and j are " + i + " and " + j);
    j = computeSomething(i);
}
```

5. (7 points) In your own words, tell me how `int` and `double` values are stored and what the limits are on the numbers that can be stored in these two formats.
6. (8 points) Assume you have a `String` variable defined as
`String s = "The word 'XXXXX' is to be written out."`
Use the built in string manipulation functions to write additional code to change the contents of the string into
`The word 'first' is to be written out.` (Simplification: You may assume that whatever is done to the string is done by changing the five character sequence of X's into a five character sequence that is a word.)
7. (10 points) Write a method
`Public double area(double radius)`
to return the area of a circle when given the radius as a parameter.

8. (15 points) Write a complete class that extends `Robot`. Your class should have an instance variable `movesTaken` that records the total number of moves taken. It should include, in addition to a constructor, an accessor and a mutator to deal with the `movesTaken` variable. You should assume that there is a method
`Public void moveRobot(int eastWest, int northSouth)`
that will move the robot as indicated by the parameters to this method. **THE ONLY PART OF THIS METHOD YOU NEED TO WRITE, OTHER THAN THE CODE THAT ACTUALLY CREATES THE METHOD, IS THE UPDATE TO THE NUMBER OF MOVES TAKEN.**
9. (10 points) Write a method
`public void moveRobot(int howMany)`
to move a robot `howMany` steps forward in the direction it currently happens to be facing. **HOWEVER**, there may be **ONE** wall in the path. You need to test for the wall, and if there is a wall in the way, your robot should take a right turn and continue for the rest of its scheduled number of moves in this different direction.
10. (5 points) What is an *enumerated* data type? Give an example of an enumerated data type that you have been using all along.