

ComS 207: Programming I
Midterm 2, Tue. Mar 21, 2006

Student Name: Ms. Perfect

Student ID Number:

Recitation Section:

1. True/False Questions (10 x 1p each = 10p)

Determine the value of each boolean expression given the following initial conditions:

boolean a=true, b=false;

int c=5, d=4;

- | | |
|---|-------|
| (a) (a (a && b)) | TRUE |
| (b) (b (a && b)) | FALSE |
| (c) ((!(a && b)) == (!a !b)) | TRUE |
| (d) (a && (c == d)) | FALSE |
| (e) (b && (c != d)) | FALSE |
| (f) ((!(a b)) == (!a && !b)) | TRUE |
| (g) (!(!a) && (c-d > 0) && (b !b)) | TRUE |
| (h) ((!false != !(true)) (3*3 + d*d <= c*c) && (a !a)) | TRUE |
| (i) (!a (!b && (c != 0))) == (!(d -13%5 -1 > 0)) && !(a!=b)) | FALSE |
| (j) ((c-3/2 == 4) && (!b && false)) (c == ++d) && (a && true)) | TRUE |

2. Short Answer Questions (5 x 2p each = 10p)

(a) What is a ragged array?

In Java a 2D array is actually represented as a 1D array of references to 1D arrays. Because the first dimension holds only references to array objects it is possible that the dimensions of these other arrays could be different. Arrays that have this property are called ragged arrays. The concept generalizes to multiple dimensions as well.

(b) What is a method?

A method is a group of programming statements that is given a name and is delimited by '{' '}' braces. When a method is invoked, its statements are executed. An object can have multiple methods associated with it.

(c) What is the difference between an if statement and a switch statement?

The expression evaluated at the beginning of a switch statement must have an integral type. In other words, it must be either an int or a char. The if statement can have any types of variables in its test expression as long as the result evaluates to a boolean variable. A switch statement can be used to test multiple conditions at the same time which sometimes produces more readable code.

(d) What is a debugger?

A debugger is a software tool which allows a programmer to check his/her program for errors/bugs. Among other things, debuggers allow step by step execution of the program as well as inspection of the values of program variables at run time.

(e) What is the difference between a do loop and a while loop?

The difference between the two loops is the time at which the loop condition is evaluated. In do loops it is evaluated at the end of the loop. In while loops it is evaluated first. Do loops are executed at least once. While loops may not be executed at all.

3. Code Snippets (3 x 5p each = 15p)

Write a code snippet (3-5 lines max) that produces the result specified below.

- (a) Print the odd numbers from 1 to 100.

The output should be in the following format: 1, 3, 5, 7, ...

```
for(int i=1; i<100; i+=2)
    System.out.print(i + ", ");

System.out.println();
```

- (b) Calculate and print the numbers between 1 and 1000 that are powers of 2.

The output should be in the following format: 2, 4, 8, 16, ...

```
for(int p=2; p< 1000; p*=2)
    System.out.print(p + ", ");

System.out.println();
```

- (c) Calculate and print the sum of the numbers between 1 and 100 (inclusive).

```
int sum=0;
for(int n=1; n<=100; n++)
    sum+=n;
System.out.println(sum);
```

4. Rewriting Code (4 x 5p each = 20p)

(a) Rewrite the following while loop as a for loop.

```
int i = 20;
while(i > -10) {
    i = i -5;
    System.out.println("Value of i is " + i);
}
```

```
// Sample Solution
for(i=20; i> -10; )
{
    i= i-5;
    System.out.println("Value of i is " + i);
}
```

(b) Rewrite the following do loop as a for loop.

```
int number= -5, j=0;
do {
    System.out.println("number= " + number);
    number+=j;
    j++;
} while(number <10);
```

```
// Sample Solution
for(j=0, number=-5; number <10; number+=j, j++)
{
    System.out.println("number= " + number);
}
```

(c) Rewrite the following switch statement using only if and else statements.

```
switch(letter)
{
    case 'A':
    case 'a':
        aCount++;
        break;
    case 'B':
    case 'b':
        bCount++;
        break;
    default:
        System.out.println("Invalid Input");
        break;
}
```

```
// Sample Solution
if((letter=='a') || (letter == 'A'))
    aCount++;
else if((letter=='b') || (letter == 'B'))
    bCount++;
else
    System.out.println("Invalid Input");
```

(d) Rewrite the following for loop as a do loop.

```
for(;;)
    ;
```

```
// Sample Solution
do
{
}while(true);
```

5. What is the Output? (2 x 5p each = 10p)

For each of the following code snippets write down what will be printed on the screen.

```
(a) for(int a=0; a<5; a++)
    {
        for(int b=0; b<a; b++)
            System.out.print("#");
        System.out.println("#\n#");
    }
```

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

```
(b) int[] a={1, 2, 3, 4, 5, 6, 7, 8, 9, 0};
    int[] b={5, 3, 1, 0, 9, 2, 6, 8, 4, 7};
    for(int i=0; i<b.length; i++)
        System.out.print(b[a[b[i]]] + ", ");
    System.out.println();
```

```
6, 9, 1, 3, 5, 0, 8, 7, 2, 4
```

6. Programming Projects (TOTAL 65p, but each has a different weight)

(a) Numbers to Text (15p)

Write a complete Java program that reads in an integer number and output its digits in reverse order but in English. For example, if the input was 608 then the output should be "Eight Zero Six". You can assume that the user will be nice and will always input an integer.

```
import java.util.Scanner;
public class NumbersToText
{
    public static void main(String[] args)
    {
        int number, digit;
        Scanner scan= new Scanner(System.in);
        System.out.print("Enter an Integer Number: ");
        number = scan.nextInt();
        do
        {
            digit = number % 10;
            number = number / 10;
            switch(digit)
            {
                case 0: System.out.print("Zero "); break;
                case 1: System.out.print("One "); break;
                case 2: System.out.print("Two "); break;
                case 3: System.out.print("Three "); break;
                case 4: System.out.print("Four "); break;
                case 5: System.out.print("Five "); break;
                case 6: System.out.print("Six "); break;
                case 7: System.out.print("Seven "); break;
                case 8: System.out.print("Eight "); break;
                case 9: System.out.print("Nine "); break;
            }
        } while (number > 0);

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

(b) 3D Vectors (15p)

Write a complete Java program which asks the user to enter the coordinates of two 3D vectors. In other words, the program must read two sets of three floating point numbers which represent the x , y , and z coordinates of the two vectors: $u = (u_x, u_y, u_z)$ and $v = (v_x, v_y, v_z)$. The program must then calculate and print on the screen the following four values associated with the vectors: 1) the length of the vector u ; 2) the length of the vector v ; 3) the dot product between the two vectors; 4) the angle θ between the two vectors.

Hint 1: The length (or norm) of the vector u is defined as: $\|u\| = \sqrt{u_x^2 + u_y^2 + u_z^2}$

Hint 2: The dot product of two vectors u and v is defined as: $u \cdot v = u_x v_x + u_y v_y + u_z v_z$

Hint 3: The angle between two vectors u and v is defined as: $\theta = \arccos\left(\frac{u \cdot v}{\|u\| \|v\|}\right)$

```
import java.util.Scanner;
public class Vectors3D
{
    public static void main(String[] args)
    {
        double[] u= new double[3];
        double[] v= new double[3];
        char[] axes={'X', 'Y', 'Z'};

        Scanner scan= new Scanner(System.in);

        for(int i=0; i < u.length; i++) {
            System.out.print("Enter " + axes[i] + "-component of vector u: ");
            u[i]=scan.nextDouble();
        }

        System.out.println();
        for(int i=0; i < v.length; i++) {
            System.out.print("Enter " + axes[i] + "-component of vector v: ");
            v[i]=scan.nextDouble();
        }

        double norm_u = 0.0;
        double norm_v = 0.0;
        double u_dot_v = 0.0;
```



```

for(int i=0; i < 3; i++) {
    norm_u += u[i]*u[i];
    norm_v += v[i]*v[i];
    u_dot_v += u[i]*v[i];
}

norm_u = Math.sqrt(norm_u);
norm_v = Math.sqrt(norm_v);

double theta = Math.acos( u_dot_v/(norm_u * norm_v));

System.out.println();
System.out.println("1) Norm of u = " + norm_u);
System.out.println("2) Norm of v = " + norm_v);
System.out.println("3) u dot v = " + u_dot_v);
System.out.println("4) Theta = " + theta*180.0/Math.PI);
}
}

```

(c) Largest Five (15p)

Write a Java program that initializes an integer array of size 10 with random integers between 1 and 10. The program must then find and print the five largest numbers stored in the array and their corresponding indices. If there are two (or more) numbers in the top five that have the same magnitude (this is possible since the array is initialized at random) then the number with the smaller index should be printed first.

```
import java.util.Random;
public class LargestFive
{
    public static void main(String[] args)
    {
        // We use two arrays. Original remains the same.
        // Copy gets overwritten during the search.
        int[] original = new int[10];
        int[] copy     = new int[10];

        Random generator = new Random();
        for(int i=0; i < original.length; i++) {
            original[i]= generator.nextInt(10)+1;
            copy[i]    = original[i];
        }

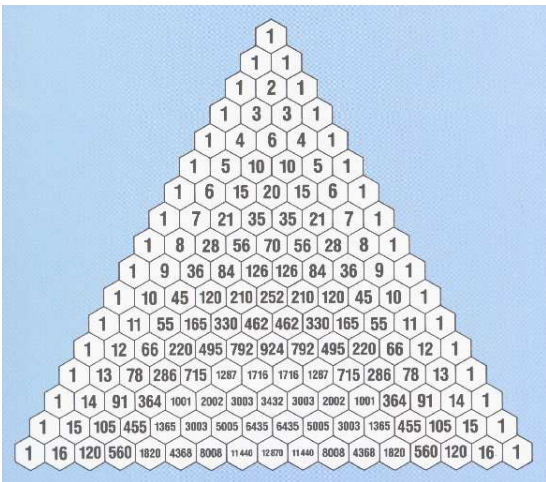
        for(int k=0; k< 5; k++)
        {
            int max=-1;
            int idx= -1;

            for(int i=0; i < copy.length; i++) {
                if(max< copy[i]) {
                    max=copy[i];
                    idx=i;
                }
            }
            System.out.println((k+1) + ") " + copy[idx] + " at index " + idx);
            copy[idx]=-1; // don't consider this number anymore
        }
    }
}
```

(d) Pascal's Triangle (20p)

Write a Java program that calculates and prints the first nine rows (0, 1, ... 8) of Pascal's Triangle (see the figure below). This triangle has many interesting mathematical properties and is formed as described below. Row 0 contains only a single number which is equal to 1. The first and the last number in all other rows is also 1. Any other number in the triangle can be calculated by adding the two numbers located above it. For example, the number 2 on row two is equal to the sum of the two 1's located above it on row one. Similarly, the first 15 on row six is equal to the sum of the numbers 5 and 10 located above it on row five. And so on.

NOTE: You will get 15 points if your program can calculate and print the first nine rows regardless of the output format. To get the extra 5 points you must print the output in the form of a triangle similar the figure below (**hint 1:** printing the output in this format is similar to printing the asterisks in problem 5.13.d on page 281 in your textbook; **hint 2:** the largest number in the first nine rows has at most two digits).



```
// Alternative way to visualize it
//
// row 0:  1
// row 1:  1  1
// row 2:  1  2  1
// row 3:  1  3  3  1
// row 4:  1  4  6  4  1
// row 5:  1  5 10 10  5  1
// row 6:  1  6 15 20 15  6  1
// row 7:  1  7 21 35 35 21  7  1
// row 8:  1  8 28 56 70 56 28  8  1
```

```

public class PascalsTriangle
{
    public static void main(String[] args)
    {
        final int DEPTH=9;
        // Initialize the Memory for the Triangle
        int[] [] P = new int[DEPTH] [];

        for(int row=0; row < DEPTH; row++)
            P[row] = new int[row+1];

        P[0][0]= 1;    // Fill in the 0-th row

        // Fill in the values of the Triangle
        for(int row=1; row<DEPTH; row++) {
            P[row][0]=1;    // First entry is always 1

            for(int col=1; col<row; col++)
                P[row][col]= P[row-1][col-1] + P[row-1][col];

            P[row][row]=1; // Last entry is always 1
        }

        // Print the Triangle
        for(int row=0; row < DEPTH; row++) {
            for(int s=0; s< DEPTH-row; s++)
                System.out.print(" "); // two spaces

            for(int col=0; col<=row; col++)
                System.out.printf("%2d ", P[row][col]);

            System.out.println();
        }
    }
}

```