CS 2420
Fall semester section 01
Ricardo Gonzalez

Review: Chapter 1

Written section:

1.2: the three types of comments used in java are single line //, multiline comment /* */, and documentation comment /** */.

1.3: the 8 primitive types in java are int, double, char, boolean, float, long, byte, and short.

1.4: the difference between * and *= operators are if you use * you are just multiplying to variable together like z*t if z=2 and t =3 then the answer is 6, but if you use the operator =* and you do z and t again like z*=t you will change the value of z instead of just outputting a number so it would be like z = z*t so you would still output 6 but now 6 would be the value for z.

1.6: the 3 types of loops in java are for loop which is mostly used if you want to loop through something a fixed number of times, then a while loop which is repeated if the conditions are meet and if not, it stops repeating, and then a do while loop which is similar to the while loop but it will execute a block of code once and then check to see if a condition is meet and if it is then it will keep repeating.

1.9 method overloading is when a class has other methods that have the same name but have different argument list.

In Theory:

1.12: first you would execute true && false which would end up being false since in the "and" boolean operation it has to be the same or it is false. But after it is false || true which in the "or" boolean operation if one is true then it is true.

1.14: that output that would come out from this line of codes would be 0, what ever input is put for variable x is what will be outputted from the print.

Programming:

1.16: first I did the addition table for the chart 0-9 my code was

```
//for addition table
//creating length of table to 9 including 0
int x = 10;
int y = 10;

//inhanced for loop to run through every iteration
for(int i = 0; i < x; i++)
{
    for(int j = 0; j < y; j++)
    {
        //for multiplication table
        int answer = i+j;
        System.out.print(answer + ", ");

    }
    System.out.println("\n");
}
```

My result for the code

Worked well expected the outcomes.

```
PS C:\Users\gonza\OneDrive\Documents\Yea
thms and Data Structures> javac Chapter1
PS C:\Users\gonza\OneDrive\Documents\Yea
thms and Data Structures> java Chapter1
0, 1, 2, 3, 4, 5, 6, 7, 8, 9,

1, 2, 3, 4, 5, 6, 7, 8, 9, 10,

2, 3, 4, 5, 6, 7, 8, 9, 10, 11,

3, 4, 5, 6, 7, 8, 9, 10, 11, 12,

4, 5, 6, 7, 8, 9, 10, 11, 12, 13,

5, 6, 7, 8, 9, 10, 11, 12, 13, 14,

6, 7, 8, 9, 10, 11, 12, 13, 14, 15,

7, 8, 9, 10, 11, 12, 13, 14, 15, 16,

8, 9, 10, 11, 12, 13, 14, 15, 16, 17,

9, 10, 11, 12, 13, 14, 15, 16, 17, 18,
```

Then I did Multiplication

```
//for multiplication table
//creating length of table to 9 including 0
int x = 10;
int y = 10;

//inhanced for loop to run through every iteration
for(int i = 0; i < x; i++)
{
    for(int j = 0; j < y; j++)
    {
        //for multiplication table
        int answer = i*j;
        System.out.print(answer + ", ");

    }
    System.out.println("\n");
}
```

Outcome is also what I expected.

```
0, 0, 0, 0, 0, 0, 0, 0, 0, 0,

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

0, 2, 4, 6, 8, 10, 12, 14, 16, 18,

0, 3, 6, 9, 12, 15, 18, 21, 24, 27,

0, 4, 8, 12, 16, 20, 24, 28, 32, 36,

0, 5, 10, 15, 20, 25, 30, 35, 40, 45,

0, 6, 12, 18, 24, 30, 36, 42, 48, 54,

0, 7, 14, 21, 28, 35, 42, 49, 56, 63,

0, 8, 16, 24, 32, 40, 48, 56, 64, 72,

0, 9, 18, 27, 36, 45, 54, 63, 72, 81,
```

1.19:

Checked the code with modulus to see if it was an int because if the reminder was zero then it is and int and it prints

```
//ran through an enhanced for loop to get through all the options for the pairs a and b
for(int a = 1; a < 1000; ++a)
{
    for(int b = 2; b < 1000; ++b)
    {
        //then used an if statement to see if the math worked
        if((a*a + b*b + 1)%(a*b) == 0)
        {
            System.out.println(a + ", " + b);
        }
    }
}
```

Here are the outcomes