CIS 36A – 3rd In Class / Take Home Assignment – **10 Points**

Student Name kachilau **Student ID** 10819338 **Point Total**

Pages 29 - 39 in ***Java Programming A Comprehensive Introduction***

Section 1: Define

control statements- Control statements may be used to control the execution sequence.

If (*condition*) statement- Conditional statements are used to perform different actions based on different conditions.

Boolean Expressions- an expression in a programming language that produces a Boolean value when evaluated, i.e. one of true or false.

Relational Operators- a programming language construct or operator that tests or defines some kind of relation between two entities.

(Define Each Symbol)

< less than

<= less than or equal to

> greater than

>= greather than or equal to

== equal to

!= not equal to

CIS 36A – 3rd In Class / Take Home Assignment – **10 Points**

Student Name kachilau **Student ID** 10819338 **Point Total**

Import statement-

(Scanner class is not explained thoroughly in the given text. Here is an online resource to hopefully make the point more clear.

http://www.cs.utexas.edu/users/ndale/Scanner.html)

What does java.util.Scanner and **new** Scanner (System.in); allow the programmer to do?

They both allow programmer or user to input a value

constructor- a class is a special type of subroutine called to create an object. It prepares the new object for use, often accepting arguments that the constructor uses to set required member variables.

Code block- a free, open source cross-platform IDE which supports multiple compilers including GCC, Clang and Visual C++.

loop- a method of control loop

for(initialization; condition; iteration)statement;

loop control variable- initialized, tested, and changed as the loop executes. It is an ordinary int variable, but it is used in a special role

CIS 36A – 3rd In Class / Take Home Assignment – **10 Points**

Student Name kachilau **Student ID** 10819338 **Point Total**

Code block- a free, open source cross-platform IDE which supports multiple compilers including GCC, Clang and Visual C++.

Import statement- Enables type names to be referenced without namespace qualification.

What does java.util.Scanner and **new** Scanner (System.in); allow the programmer to do?

They both allow programmer or user to input a value

Programming Assignments

Task 1 –

Page 39. **in *Programming A Comprehensive Introduction***

Update your program from Assignment 2, Task #2

Allow the user to input their weight for the earth weight to moon weight conversion problem. Add an **if** statement that prompts the user if she inputs 0 or a negative number for her earth weight.

#13. The moon's gravity is about 17 percent that of the earth's. (Meaning you weigh less on the moon). Write a program that computes your affective weight on the moon.

CIS 36A – 3rd In Class / Take Home Assignment – **10 Points**

Student Name kachilau **Student ID** 10819338 **Point Total**

```java
package javaapplication3;

import java.util.Scanner;

public class JavaApplication3 {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {

       Scanner input = new Scanner(System.in);
        //input can be any name

        double yourweight;

        double moonweight;

        System.out.print("Enter your weight: ");
        yourweight = input.nextDouble();

        moonweight = yourweight * (17.0/100.0);

        System.out.println("Your weight on the moon is " + moonweight);
    }

}
```

JavaApplication3 > main >

ut - JavaApplication3 (run)

```
run:
Enter your weight: 16
Your weight on the moon is 2.72
BUILD SUCCESSFUL (total time: 2 seconds)
```

CIS 36A – 3rd In Class / Take Home Assignment – **10 Points**

Student Name kachilau **Student ID** 10819338 **Point Total**

Task 2-

Page 39. **in *Programming A Comprehensive Introduction***

#14 (modified)

Adapt Try This 1-2 so that it prints a conversion table of inches to meters. Display **3 feet** of conversions, inch by inch. **(36 inch to meter conversions)** Output a blank line every 12 inches. (One meter equals approximately 39.37 inches.)

CIS 36A – 3rd In Class / Take Home Assignment – **10 Points**

Student Name kachilau **Student ID** 10819338 **Point Total**

```
7   package javaapplication1;
8
9   import java.util.Scanner;
10
11  public class JavaApplication1 {
12
13      /**
14       * @param args the command line arguments
15       */
16      public static void main(String[] args) {
17          // TODO code application logic here
18
19          Scanner input = new Scanner(System.in);
20
21          int line;
22          double count;
23
24          line = 0;
25          for (count = 1.0; count < 37; count++) {
26              System.out.println(count + " inches = " + count / 39.37 + " meters ");
27
28              line++;
29              if (line == 12) {
30                  System.out.println("");
31                  line = 0;
32              }
33
34          }
35      }
36  }
37
```

CIS 36A – 3rd In Class / Take Home Assignment – **10 Points**

Student Name kachilau **Student ID** 10819338 **Point Total**

```
run:
1.0 inches = 0.025400050800101603 meters
2.0 inches = 0.05080010160020321 meters
3.0 inches = 0.07620015240030481 meters
4.0 inches = 0.10160020320040641 meters
5.0 inches = 0.12700025400050802 meters
6.0 inches = 0.15240030480060962 meters
7.0 inches = 0.17780035560071122 meters
8.0 inches = 0.20320040640081283 meters
9.0 inches = 0.2286004572009144 meters
10.0 inches = 0.25400050800101603 meters
11.0 inches = 0.27940055880111764 meters
12.0 inches = 0.30480060960121924 meters

13.0 inches = 0.33020066040132084 meters
14.0 inches = 0.35560071120142245 meters
15.0 inches = 0.38100076200152405 meters
16.0 inches = 0.40640081280162565 meters
17.0 inches = 0.43180086360172726 meters
18.0 inches = 0.4572009144018288 meters
19.0 inches = 0.4826009652019304 meters
20.0 inches = 0.5080010160020321 meters
21.0 inches = 0.5334010668021336 meters
22.0 inches = 0.5588011176022353 meters
23.0 inches = 0.5842011684023368 meters
24.0 inches = 0.6096012192024385 meters

25.0 inches = 0.63500127000254 meters
26.0 inches = 0.6604013208026417 meters
27.0 inches = 0.6858013716027432 meters
28.0 inches = 0.7112014224028449 meters
29.0 inches = 0.7366014732029464 meters
30.0 inches = 0.7620015240030481 meters
31.0 inches = 0.7874015748031497 meters
32.0 inches = 0.8128016256032513 meters
33.0 inches = 0.8382016764033529 meters
34.0 inches = 0.8636017272034545 meters
35.0 inches = 0.8890017780035561 meters
36.0 inches = 0.9144018288036576 meters

BUILD SUCCESSFUL (total time: 0 seconds)
```

Task 3-

Pg. 41 # 29

CIS 36A – 3rd In Class / Take Home Assignment – **10 Points**

Student Name kachilau **Student ID** 10819338 **Point Total**

Write a program that prints out the first 20 squared numbers**.** Starting at x = 1 until x = 20. **Do not use java.lang.Math**

Create a line counter variable that inputs a blank line every 5 lines.

Display outputs as shown below.

1 squared = 1

2 squared = 4

3 squared = 9

4 squared = 16

// Every 5 lines create a blank line.

5 squared = 25

6 squared = 36

7 squared = 49

Continued….

CIS 36A – 3rd In Class / Take Home Assignment – **10 Points**

Student Name kachilau **Student ID** 10819338 **Point Total**

```
7   package javaapplication1;
8
9   import java.util.Scanner;
10
11  public class JavaApplication1 {
12
13      /**
14       * @param args the command line arguments
15       */
16      public static void main(String[] args) {
17          // TODO code application logic here
18
19          Scanner input = new Scanner(System.in);
20
21          int line;
22          int count;
23
24          line = 0;
25          for (count = 1; count < 21; count++) {
                System.out.println(count + " squared = " + count * count);
27
28              line++;
29              if(line == 5) {
30                  System.out.println("");
31                  line = 0;
32              }
33          }
34      }
35  }
```

CIS 36A – 3rd In Class / Take Home Assignment – **10 Points**

Student Name kachilau **Student ID** 10819338 **Point Total**

```
1 squared = 1
2 squared = 4
3 squared = 9
4 squared = 16
5 squared = 25

6 squared = 36
7 squared = 49
8 squared = 64
9 squared = 81
10 squared = 100

11 squared = 121
12 squared = 144
13 squared = 169
14 squared = 196
15 squared = 225

16 squared = 256
17 squared = 289
18 squared = 324
19 squared = 361
20 squared = 400

BUILD SUCCESSFUL (total time: 0 seconds)
```