Table of Contents

Osmond's Academic Portfolio

Taipei Municipal Zhong-Lun High School

Course Name
Basic Java Programming

Instructor

Dr. Ah-Fur Lai

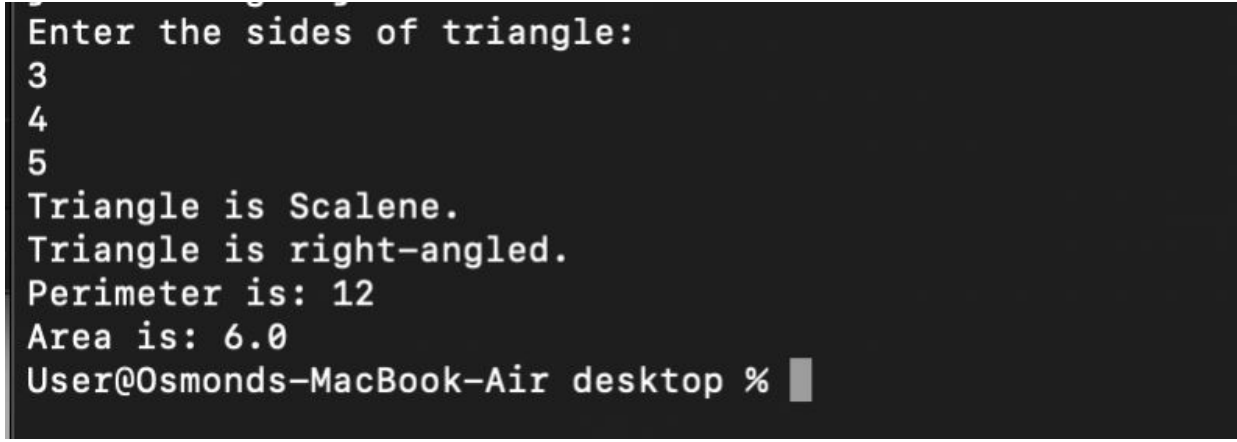
Brief Introduction
This is a course where we get to learn how to program and write codes using Java, learning the principles of computer science and programming.

I. Motivation & Learning Objectives

I’ve always been interested in modern technology and computer science and so I attended this course with strong passion and momentum. In the very beginning, I knew and set goals that I wanted to pay a lot of efforts and get familiar with Java through the course after one semester of learning. In this course, we learned different syntax, loops, and different algorithms on each lesson. Also, to see if we actually learned something during this remote course, there were online quizzes each lesson and two main term tests held in the middle and the end of the semester. In addition, for the purpose of gettin used to coding on our own, there were coding assignments about what we’ve been taught after each lesson as well. In the end, learning computer science can also help me develop better ability of problem solving and critical thinking since a core idea in programming is just to help people fix problems and make the world a better place.

II. Task/Project & Learning Methods (A)-Triangle Calculator(1)

This coding assignment is a triangle calculator that helps us to identify the types of triangle and results the perimeter and area through given side lengths of a triangle. I coded this beginner-friendly program as a practice for me to know how to use import scanner class and make a good use of math.sqrt. From this task, I learned that Java is an object-oriented programming language, and I always have to import scanner class first in order to let users input information.



II. Task/Project & Learning Methods (A)-Triangle Calculator(2)

Source code:Triangle.java

```
import java.util.*;
class Triangle
{
static String triangleType(int a, int b, int c) //function to find
type of triangle
{
    if (a == b && b == c)
        return "Equilateral";
    else if (a == b || b == c || a == c)
        return "Isosceles";
    else
        return "Scalene";

}
static String angleType(int a, int b, int c) //function to find
angle
{
int x = a * a;
int y = b * b;
int z = c * c;
```

```
if(a>b && a>c) // when largest side is 'a'
{
    if(x == y + z)
    return "right-angled";
    else if(x < y + z)
    return "acute-angled";
    else
    return "obtuse-angled";

}
else if(b>a && b>c) // when largest side is 'b'
{
   if(y == x + z)
    return "right-angled";
    else if(y < x + z)
    return "acute-angled";
    else
    return "obtuse-angled";
}
```

II. Task/Project & Learning Methods (A)-Triangle Calculator(3)

Source code:

```
else // when largest side is 'c'
{
    if(z == x + y)
    return "right-angled";
    else if(z < x + y)
    return "acute-angled";
    else
    return "obtuse-angled";

}
}
static void triangle(int a, int b, int c)
{
System.out.println("Triangle is " + triangleType(a, b, c)+".");
System.out.println("Triangle is " + angleType(a, b, c)+".");
}
```

```
public static void main(String[] args)
{
    int a, b, c;
    System.out.println("Enter the sides of triangle: ");
    Scanner sc = new Scanner(System.in);
    a = sc.nextInt();
    b = sc.nextInt();
    c = sc.nextInt();
    if(a == 0 || b == 0 || c == 0) // to check whether length of
any side is 0 or not
      System.out.println("Triangle cannot be formed");
    else
      triangle(a, b, c);
int s=(a+b+c);
int k=(a+b+c)/2 ;
System.out.println("Perimeter is: "+s);
double h=k*(k-a)*(k-b)*(k-c);
double X=Math.sqrt(h);
System.out.println("Area is: "+X);
}
}
```

II. Task/Project & Learning Methods (B)-GCD(1)

GCD, greatest common divisor, which is the greatest common divisor that multiple numbers shared. In this task, I wrote the program that can help people find GCD of two integers using the while loop. From this task, I basically just learned how a loop works in Java and when to use it.

II. Task/Project & Learning Methods (B)-GCD(2)

Source code: gcd.java

```
import java.util.Scanner;
public class gcd{
public static void main(String[]args){

Scanner inp= new Scanner(System.in);

while(true){
int gcd=1, i=1;
System.out.println("Type in two integers to find their greatest common divisor: ");

int n1=inp.nextInt();
int n2=inp.nextInt();
do{

if(n1%i==0 && n2%i==0)
gcd=i;
i++;

}while(i<=n1 && i<=n2);

System.out.println("GCD of "+n1+" and "+n2+" is "+ gcd+".");
}

}//main
}//class
```

II. Task/Project & Learning Methods (C)-Phrases game(1)

This is a program that helps English learners to learn English phrases.
I designed the program using two strings and a for loop. In particular, I used the "indexOf" method. With that method, I'm able to print out certain words or characters in a string. Therefore, the "indexOf" string helps me to print out the hints of the phrases. That said, I'm still not sure how to correctly print using the random method to pick out random number with no duplicated. Also, I shared the program with my twin sisters, as well as teaching them some English phrases along the way. This is probably the best about programming that you get to help others and share happiness and knowledge.

II. Task/Project & Learning Methods (C)-Phrases game(2)

HOW IT GOES:

```
User@Osmonds-MacBook-Air desktop % javac phrase_2.java
User@Osmonds-MacBook-Air desktop % java phrase_2.java
This is a game where you can learn some English phrases.
You will get 10 points for each correct answer.
Definition: To go to bed or go to sleep
Hint: Starts with 'h' and ends with 'k'
Enter the corresponding phrase: hit the sack
Correct!

Definition: It's better to do something late than not at all
Hint: Starts with 'b' and ends with 'r'
Enter the corresponding phrase: better late than never
Correct!

Definition: To decide to stop working for the day
Hint: Starts with 'c' and ends with 'y'
Enter the corresponding phrase: call it a day
Correct!

Definition: Feeling unwell or sick
Hint: Starts with 'u' and ends with 'r'
Enter the corresponding phrase: under the
Incorrect!

Definition: Good luck
Hint: Starts with 'b' and ends with 'g'
Enter the corresponding phrase: break a leg
Correct!

Game over! Your final score is: 40
Do you want to play again? (yes/no): yes
Definition: To go to bed or go to sleep
Hint: Starts with 'h' and ends with 'k'
Enter the corresponding phrase: hit the sack
Correct!

Definition: It's better to do something late than not at all
Hint: Starts with 'b' and ends with 'r'
Enter the corresponding phrase: better late than never
Correct!

Definition: To decide to stop working for the day
Hint: Starts with 'c' and ends with 'y'
Enter the corresponding phrase: call it a day
Correct!

Definition: Feeling unwell or sick
Hint: Starts with 'u' and ends with 'r'
Enter the corresponding phrase: under the weather
Correct!

Definition: Good luck
Hint: Starts with 'b' and ends with 'g'
Enter the corresponding phrase: break a leg
Correct!

Game over! Your final score is: 50
Do you want to play again? (yes/no): no
Thank you for playing the English Phrases Game!
User@Osmonds-MacBook-Air desktop %
```

II. Task/Project & Learning Methods (C)-Phrases game(3)

Source code: phrase_2.java

```
import java.util.Scanner;
import java.util.Random;

public class phrase_2{
    private static final String[] phrases = {
        "hit the sack",
        "better late than never",
        "call it a day",
        "under the weather",
        "break a leg"
    };

    private static final String[] definitions = {
        "To go to bed or go to sleep. Hint: Starts with 'h' and ends with 'k'",
        "It's better to do something late than not at all. Hint: Starts with 'b' and ends with 'r'",
        "To decide to stop working for the day. Hint: Starts with 'c' and ends with 'y'",
        "Feeling unwell or sick. Hint: Starts with 'u' and ends with 'r'",
        "Good luck. Hint: Starts with 'b' and ends with 'g'"
    };

    public static void main(String[] args) {

//the random number generator with no duplicated
```

```
Random random=new Random();
int[]a=new int[5];
int number;

                for (int i=0; i<a.length; i++){

                a[i]=random.nextInt(5);

 for(int j=0; j<i;j++){
 if(a[i]==a[j]){

i--;
break;

}//if
}//for2
}//for1
Scanner scanner = new Scanner(System.in);
        boolean playAgain = true;

        System.out.println("This is a game where you can learn some English phrases.");
        System.out.println("You will get 10 points for each correct answer.");
```

II. Task/Project & Learning Methods (C)-Phrases game(4)

Source code: phrase_2.java

```
while (playAgain) {
        int score = 0;

        for (int i = 0; i < phrases.length; i++) {
            String phrase = phrases[i];
            String definition = definitions[i];
            String hint = definition.substring(definition.indexOf("Hint: ") + 6);
            definition = definition.substring(0, definition.indexOf(". Hint:"));

            System.out.println("Definition: " + definition);
            System.out.println("Hint: " + hint);
            System.out.print("Enter the phrase: ");
            String userInput = scanner.nextLine();

            if (userInput.equalsIgnoreCase(phrase)) {
                System.out.println("Correct!");
                score += 10;
            } else {
                System.out.println("Incorrect!");
            }

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

```
System.out.println("Game over! Your final score is: " + score);
        System.out.print("Do you want to play again? (yes/no): ");
        String playAgainInput = scanner.nextLine();

        if (!playAgainInput.equalsIgnoreCase("yes")) {
            playAgain = false;
        }
    }
```

III. Reflection

I truly enjoy this one-semester course since I actually learned how to code using Java from scratch. Not only did I develop solid background knowledge of computer science, but it also strengthened my dream of becoming a software engineer in the future for the purpose of using my own programming skills and knowledge to help others and make world a better place. Nonetheless, after completing so many coding assignments on my own, it dawned on me that software development and engineers are actually the wizards of modern technology-based society. Everyone is somehow affected and benefited from software in lots of different aspects of life. For instance, smart phones, electrical cars, computers, and so on. All these things we use in our day to day life are formed and made by software and hardware engineers. Thus, I want to devote myself into the industry and make some impacts.

IV. Difficulties/Challenges

Nothing is easy in life, and so is coding. Debugging is probably the most annoying thing that I face when it comes to coding. Plus, it's so time consuming whenever I have to cope with errors of codes. Java is a case sensitive language, so case of words matter a lot in codes, for instance, "String" and "string" is way different in Java. Thus, bugs will come up if the case is not used correctly. Another challenge I struggle with is the curly brace{}. In Java, there's rule that we have to type our codes inside curly braces, and it's always needed to wrap our main method, loops, and even methods. All in all, there are still countless I'll have to learn on the path of coding; regardless, I know that it's important to stick with my dreams despite all the difficulties and long-drawn-out debugging. It takes time to master anything. So even if it takes time and patience along the way, I have to be persistent. To sum up, coding is a marathon, not a sprint.

V. Feedback & Evaluation (from peers and teachers)

This was a remote course where we went to computer classroom and had a online lesson with the professor every Friday. As a result, I didn't really have any chance to get to know my peers. However, the professor did give me some pieces of advice of coding. We all know that the simpler, the better. And the same mindset also works the same in coding. Just like writers writing an article, they try their best to get rid of redundancy. At the same time, software engineers always try their best to make their codes look simpler as well. Of course, different complexity of codes can function the same; however, they take different amount of time. Nonetheless, practice might not make perfect in coding. Even if you spend all the time in solving coding problems on Leetcode, it doesn't guarantee that you coded "perfect". In other words, there's no perfect code; yet, each code can show different personality and specialty.