

1. Coin Dispenser
2. Sum the Digits in an Integer
3. Calculating the Body Mass Index (BMI)
4. Process First and Last Letter
5. Rock-Paper-Scissors-Lizard-Spock
6. Temperature
7. Highest Scoring Student
8. Calculate Restaurant Check

Coin Dispenser

APPLICATION DESCRIPTION:

The application will ask the user to enter the amount of change under \$1. The app uses this amount to separate and dispense the appropriate number of coins for each: quarters, dimes, nickels, and pennies. It uses the variable "change" to track the remaining change during the process.

Coin Dispenser (pseudo code)

OUTPUT: "How much change will the customer get back (under \$1)?"

INPUT: integer returnChange

integer change = 0

integer quarters = returnChange/25

change = returnChange % 25

integer dimes = change/10

change = change % 10

integer nickels = left/5

change = change % 5

integer pennies = change

OUTPUT: "The customer receives: \n"

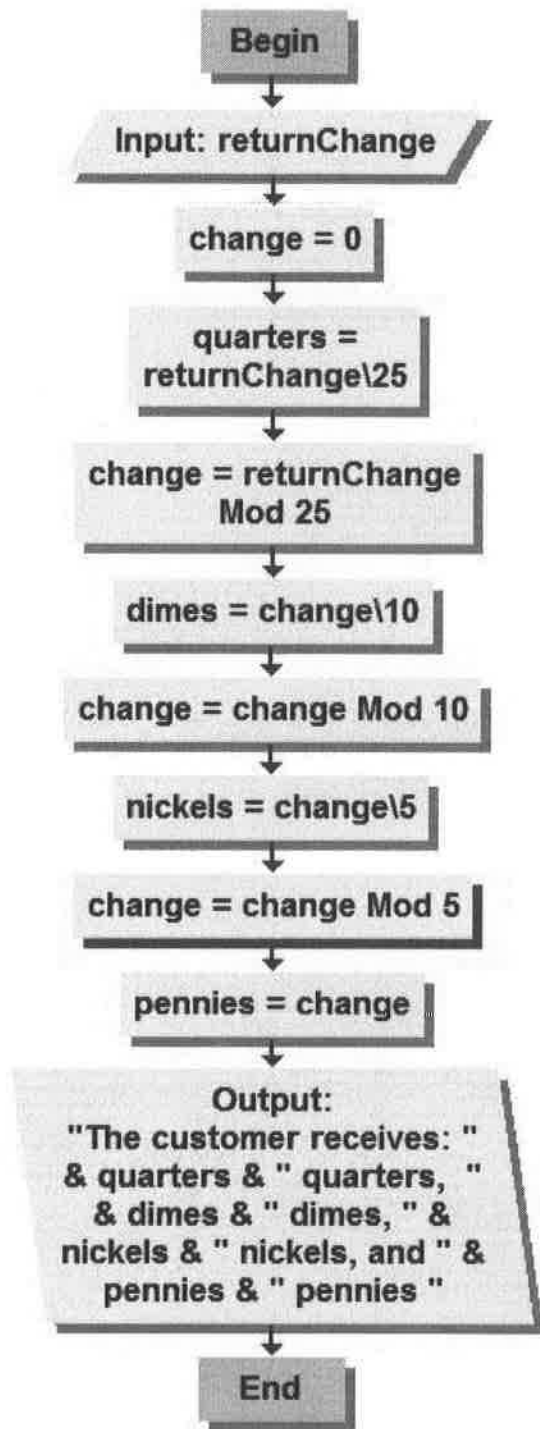
+ quarters + " quarters \n"

+ dimes + " dimes \n"

+ nickels + " nickels \n"

+ pennies + " pennies \n\n"

+ ** App Complete **



```
package labs;
import java.util.Scanner;

public class Lab1_3 {
    /* ██████████
    * CSCI 1301 T/Th -Marietta
    * Coin Dispenser
    * Dispenses the correct coins for the change (under $1) due
    */

    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        System.out.print(
            "How much change will the customer get back (under $1)? ");
        int returnChange = input.nextInt();
        int change, quarters, dimes, nickels, pennies;

        quarters = returnChange / 25; change = returnChange % 25;
        dimes = change / 10; change = change % 10;
        nickels = change / 5; change = change % 5;
        pennies = change;

        String strOut = "\nThe customer receives: \n" + quarters + " quarters\n"
            + dimes + " dimes\n" + nickels + " nickels\n"
            + pennies + " pennies\n\n** App Complete **";
        System.out.println(strOut);
        input.close();
    }
}
```

Output:

How much change will the customer get back (under \$1)? 99

The customer receives:

3 quarters

2 dimes

0 nickels

4 pennies

** App Complete **

How much change will the customer get back (under \$1)? 54

The customer receives:

2 quarters

0 dimes

0 nickels

4 pennies

** App Complete **

How much change will the customer get back (under \$1)? 66

The customer receives:

2 quarters

1 dimes

1 nickels

1 pennies

** App Complete **

How much change will the customer get back (under \$1)? 39

The customer receives:

1 quarters

1 dimes

0 nickels

4 pennies

** App Complete **

How much change will the customer get back (under \$1)? 17

The customer receives:

0 quarters

1 dimes

1 nickels

2 pennies

** App Complete **

Sum the digits in an integer

APPLICATION DESCRIPTION:

The user is asked for a three digit number. The application will separate the digits in that three digit number and add them together. Then it will convert the user's three digit number into its binary, octal, and hexadecimal equivalents.

Sum the digits in an integer (pseudocode)

OUTPUT: What is the three digit number?

INPUT: int userNumber

```
int ones = userNumber % 10
int tens = (userNumber/10) % 10
int hundreds = userNumber/100
```

```
int result = ones+tens+hundreds
```

```
OUTPUT: "\nIn dec("+userNumber+"): "+hundreds+" + "+tens+" + "
        +ones+" = "+result);
```

String orig

```
orig = Integer.toString(userNumber, 2)
```

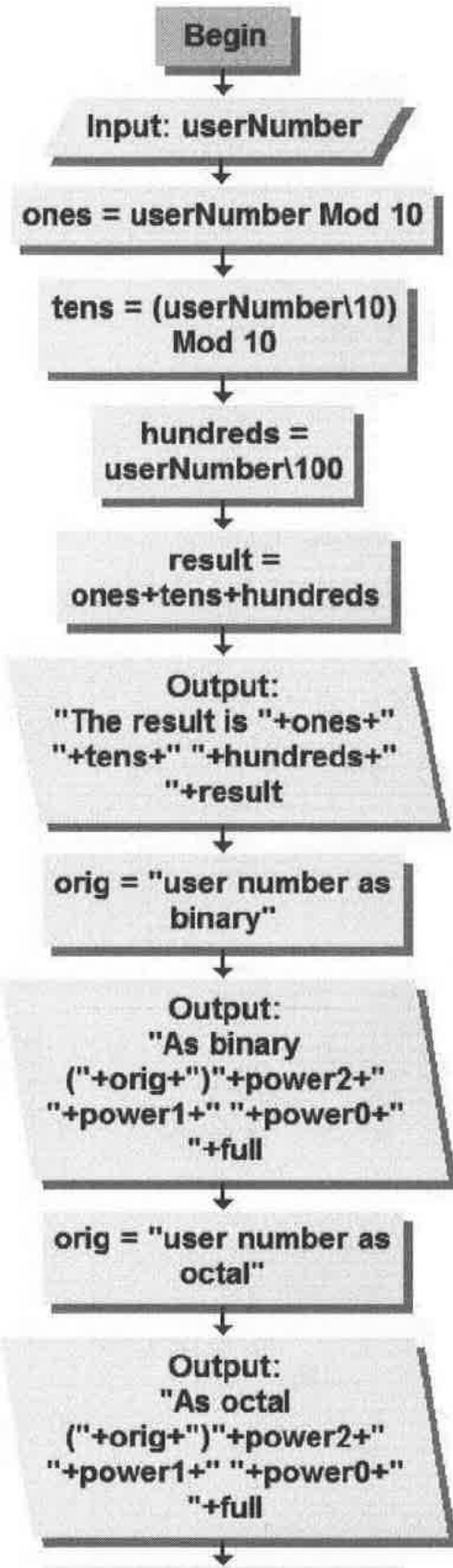
```
OUTPUT: "\nIn binary("+orig+") "
```

```
orig = Integer.toString(userNumber, 8)
```

```
OUTPUT: "\nIn octal("+orig+") "
```

```
orig = Integer.toString(userNumber, 16)
```

```
OUTPUT: "\nIn hex("+orig+") "
```

↓
orig = "user number as
hex"

↓
Output:
"As hex
("+orig+)" +power2+ "
"+power1+" "+power0+ "
"+full

↓
End

```
package labs;
import java.util.Scanner;

public class Lab2_2 {
    /* ██████████
    * CSCI 1301 T/Th -Marietta
    * Sum the digits in an integer
    * Add up the three digits in the three digit integer given by user.
    * Then gives binary, octal, and hex equivalents of that given number.
    */

    public static void main(String[] args) {
        Scanner inInt = new Scanner (System.in);
        System.out.print("What is the three digit number? ");
        int userNumber = inInt.nextInt();
        int ones = userNumber % 10
        int tens = (userNumber/10) % 10, hundreds = userNumber / 100;
        int result = ones+tens+hundreds;
        System.out.println("\nIn dec("+userNumber+"): "+hundreds+" + "
            +tens+" + "+ones+" = "+result);

        String orig;
        orig = Integer.toString(userNumber, 2);
        System.out.println("\nIn binary("+orig+"");

        orig = Integer.toString(userNumber, 8);
        System.out.println("\nIn octal("+orig+"");

        orig = Integer.toString(userNumber, 16);
        System.out.println("\nIn hex("+orig+"");

        inInt.close();
    }
}
```

OUTPUT:

What is the three digit number? 123

In dec(123): $1 + 2 + 3 = 6$

In binary(1111011)

In octal(173)

In hex(7b)

What is the three digit number? 842

In dec(842): $8 + 4 + 2 = 14$

In binary(1101001010)

In octal(1512)

In hex(34a)

What is the three digit number? 165

In dec(165): $1 + 6 + 5 = 12$

In binary(10100101)

In octal(245)

In hex(a5)

What is the three digit number? 888

In dec(888): $8 + 8 + 8 = 24$

In binary(1101111000)

In octal(1570)

In hex(378)

What is the three digit number? 987

In dec(987): $9 + 8 + 7 = 24$

In binary(1111011011)

In octal(1733)

In hex(3db)

Calculating the Body Mass Index (BMI)

APPLICATION DESCRIPTION:

The user is asked whether the input will be in U.S. measurement or Metric. If U.S. (option 1) is selected, the user is prompted for the weight and height (in feet and inches) of the individual.

The weight and height are converted to Metric, the BMI is given and interpreted. If Metric (option 2) is selected, the user is prompted for the weight and height in kilograms and meters.

The weight and height are converted to U.S. measurement, and the BMI is given and interpreted. If an option other than the given two is selected, the user is notified that it is not a valid option.

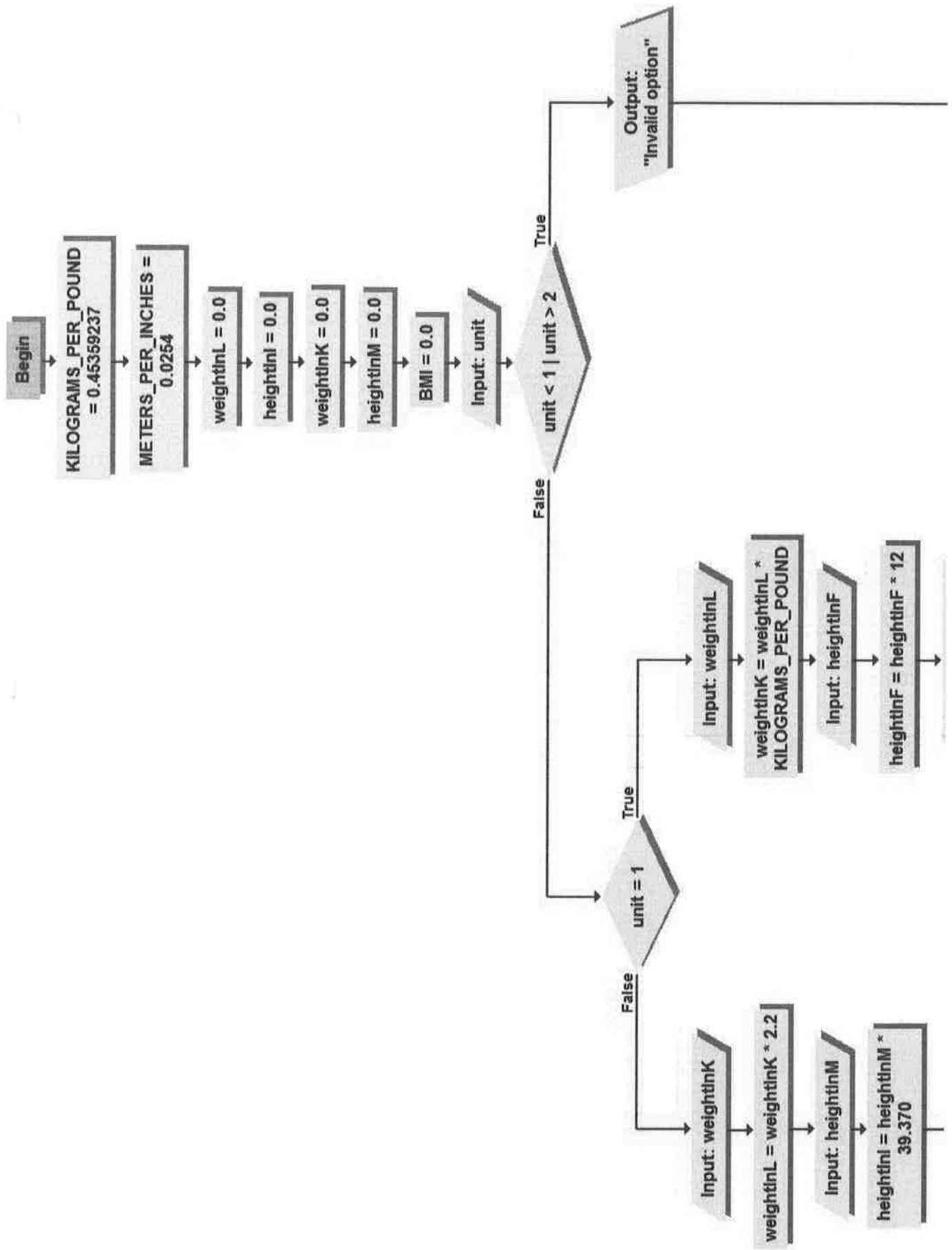
Calculating the Body Mass Index (BMI) (pseudocode)

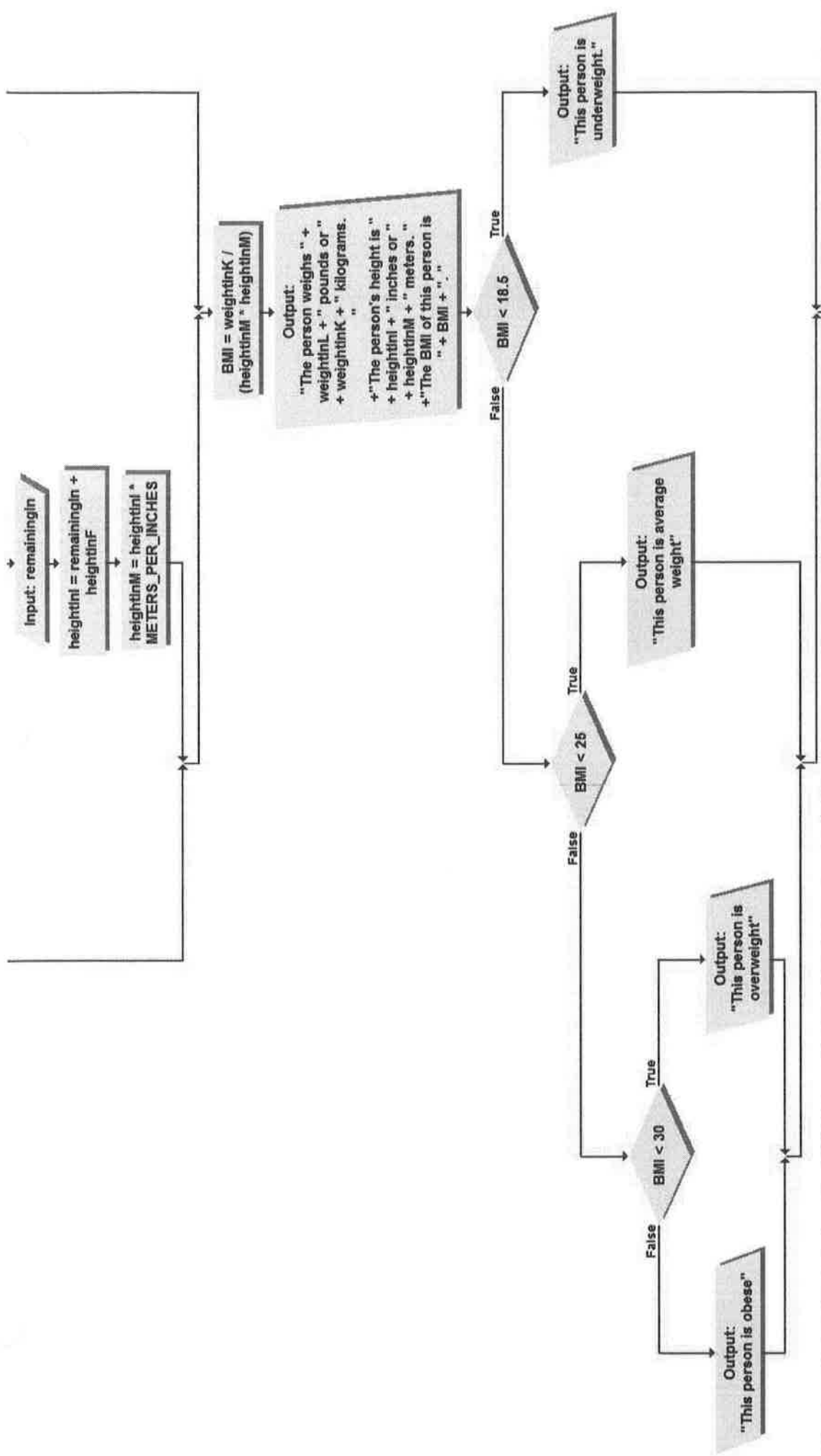
```
final KILOGRAMS_PER_POUND = 0.45359237;
final METERS_PER_INCHES = 0.0254;
weightInL
heightInI
weightInK
heightInM
BMI

OUTPUT: "Will the entered information be in U.S.(1) or Metric (2)?"
INPUT: unit

if (unit < 1 || > 2){
    OUTPUT: "Invalid Option"
}
else {
    if (unit == 1){
        OUTPUT: "How much does this person weigh (lbs only)?"
        INPUT: weightInL
        weightInK = weightInL * KILOGRAMS_PER_POUND
        OUTPUT: "How tall is the person in feet?"
        INPUT: heightInF as int
        heightInF *= 12
        OUTPUT: "any remaining inches?"
        INPUT: int remainingIn
        heightInI = (double)remainingIn + (double)heightInF
        heightInM = heightInI * METERS_PER_INCHES
    }
    else {
        OUTPUT: "How many kg does the person weigh?"
        INPUT: weightInK
        weightInL = weightInK * 2.2
        OUTPUT: "How tall is the person in meters?"
        INPUT: heightInM
        heightInI = heightInM * 39.370
    }
    BMI = weightInK/(Math.pow(heightInM, 2))
    OUTPUT "\nThe person weighs " + weightInL + " pounds or "
        + weightInK + " kilograms\n"
        +"The person's height is " + heightInI + " inches or "
        + heightInM + " meters\n"
        +"The BMI of this person is " + BMI;

    strOut = ""
    if (BMI < 18.5){strOut += "\nThis person is underweight."}
    else if (BMI < 25){strOut += "\nThis person is average weight."}
    else if (BMI < 30){strOut += "\nThis person is overweight."}
    else {strOut += "\nThis person is obese."}
    OUTPUT: strOut
    OUTPUT: "\n\n *** App Complete ***"
}
}
```





```

    Input: remainingIn
    heightInF = remainingIn /
    heightInF
    heightInM = heightInF *
    METERS_PER_INCHES
  
```

$$BMI = \frac{weightInK}{heightInM * heightInM}$$

Output:
 "The person weighs " +
 weightInL + " pounds or "
 + weightInK + " kilograms."
 +"The person's height is "
 + heightInF + " inches or "
 + heightInM + " meters."
 +"The BMI of this person is "
 + BMI + " ."

BMI < 30

Output:
 "This person is
 overweight"

BMI < 25

Output:
 "This person is average
 weight"

BMI < 18.5

Output:
 "This person is
 underweight."


```

package labs;
import java.util.Scanner;

public class Lab3_3 {
    /* ██████████
    * CSCI 1301 T/Th -Marietta
    * Calculating the Body Mass Index (BMI).
    * Calculates BMI using weight and height and converts weight and height
    * into the other measurement
    */

    public static void main(String[] args) {
        final double KILOGRAMS_PER_POUND = 0.45359237;
        final double METERS_PER_INCHES = 0.0254;
        Double weightInL, heightInI, weightInK, heightInM, BMI;

        Scanner input = new Scanner(System.in);
        System.out.print("Will the entered information be in U.S. (1) or
        Metric (2)? ");
        int unit = input.nextInt();
        if (unit < 1 || unit > 2) {System.out.print("Invalid Option");}
        else {
            if (unit == 1) {
                System.out.print("How much does this person weigh (lbs only)? ");
                weightInL = input.nextDouble();
                weightInK = weightInL * KILOGRAMS_PER_POUND;
                System.out.print("How tall is the person in feet? ");
                int heightInF = input.nextInt()*12;
                System.out.print("any remaining inches? ");
                heightInI = (double)input.nextInt()+ (double)heightInF;
                heightInM = heightInI * METERS_PER_INCHES;
            }
            else {
                System.out.print("How many kg does the person weigh? ");
                weightInK = input.nextDouble();
                weightInL = weightInK * 2.2;
                System.out.print("How tall is the person in meters? ");
                heightInM = input.nextDouble();
                heightInI = heightInM * 39.370;
            }
            BMI = weightInK/(Math.pow(heightInM, 2));
            String strOut = "\nThe person weighs " + weightInL + " pounds or "
                + weightInK + " kilograms\n" + "The person's height is "
                + heightInI + " inches or " + heightInM + " meters\n"
                + "The BMI of this person is " + BMI;
            System.out.println(strOut);

            System.out.println();
            strOut = "";
            if (BMI < 18.5) {strOut ="This person is underweight.";}
            else if (BMI < 25) {strOut ="This person is average weight.";}
            else if (BMI < 30) {strOut ="This person is overweight.";}
            else {strOut ="This person is obese.";}
            System.out.println(strOut + "\n *** App Complete ***");
        }

        input.close();
    }
}

```

OUTPUT:

Will the entered information be in U.S. (1) or Metric (2)? 1
How much does this person weigh (lbs only)? 95
How tall is the person in feet? 5
any remaining inches? 10

The person weighs 95.0 pounds or 43.09127515 kilograms
The person's height is 70.0 inches or 1.778 meters
The BMI of this person is 13.630940829738801

This person is underweight.
*** App Complete ***

Will the entered information be in U.S. (1) or Metric (2)? 1
How much does this person weigh (lbs only)? 140
How tall is the person in feet? 5
any remaining inches? 10

The person weighs 140.0 pounds or 63.502931800000006 kilograms
The person's height is 70.0 inches or 1.778 meters
The BMI of this person is 20.087702275404553

This person is average weight.
*** App Complete ***

Will the entered information be in U.S. (1) or Metric (2)? 1
How much does this person weigh (lbs only)? 95.5
How tall is the person in feet? 4
any remaining inches? 2

The person weighs 95.5 pounds or 43.318071335 kilograms
The person's height is 50.0 inches or 1.27 meters
The BMI of this person is 26.857257942215885

This person is overweight.
*** App Complete ***

Will the entered information be in U.S. (1) or Metric (2)? 1
How much does this person weigh (lbs only)? 140
How tall is the person in feet? 4
any remaining inches? 2

The person weighs 140.0 pounds or 63.502931800000006 kilograms
The person's height is 50.0 inches or 1.27 meters
The BMI of this person is 39.37189645979292

This person is obese.
*** App Complete ***

Will the entered information be in U.S. (1) or Metric (2)? 2

How many kg does the person weigh? 43.0912
How tall is the person in meters? 1.7780

The person weighs 94.80064000000002 pounds or 43.0912 kilograms
The person's height is 69.99986 inches or 1.778 meters
The BMI of this person is 13.630917057752482

This person is underweight.
*** App Complete ***

Will the entered information be in U.S. (1) or Metric (2)? 2
How many kg does the person weigh? 63.5029
How tall is the person in meters? 1.7780

The person weighs 139.70638 pounds or 63.5029 kilograms
The person's height is 69.99986 inches or 1.778 meters
The BMI of this person is 20.087692216200757

This person is average weight.
*** App Complete ***

Will the entered information be in U.S. (1) or Metric (2)? 2
How many kg does the person weigh? 43.3180
How tall is the person in meters? 1.2700

The person weighs 95.2996 pounds or 43.318 kilograms
The person's height is 49.9999 inches or 1.27 meters
The BMI of this person is 26.857213714427427

This person is overweight.
*** App Complete ***

Will the entered information be in U.S. (1) or Metric (2)? 2
How many kg does the person weigh? 63.5029
How tall is the person in meters? 1.2700

The person weighs 139.70638 pounds or 63.5029 kilograms
The person's height is 49.9999 inches or 1.27 meters
The BMI of this person is 39.37187674375348

This person is obese.
*** App Complete ***

Will the entered information be in U.S. (1) or Metric (2)? 0
Invalid Option

Will the entered information be in U.S. (1) or Metric (2)? 3
Invalid Option

Process First and Last letters

APPLICATION DESCRIPTION:

The user is asked to give a single word string. The first and last letters of the word are processed, returning the Unicode, binary, and decimal values for each of the two letters.

PSEUDOCODE:

OUTPUT: "Please enter a one word string to process."

INPUT: string str

chr = (first char of str)

chr2 = (last char of str)

uni = (chr converted to integer)

binary = uni Mod 2

divide = uni/2

```
if (divide != 0) {
    binary = (uni Mod 2) + binary
    divide = uni/2
}
```

```
if (divide != 0) {
    binary = (uni Mod 2) + binary
    divide = uni/2
}
```

```
if (divide != 0) {
    binary = (uni Mod 2) + binary
    divide = uni/2
}
```

```
if (divide != 0) {
    binary = (uni Mod 2) + binary
    divide = uni/2
}
```

```
if (divide != 0) {
    binary = (uni Mod 2) + binary
    divide = uni/2
}
```

```
if (divide != 0) {
    binary = (uni Mod 2) + binary
    divide = uni/2
}
```

```
if (divide != 0) {
    binary = (uni Mod 2) + binary
    divide = uni/2
}
```

holdDiv = uni Mod 16

hex = ""

```
if (holdDiv > 9){
    hex = 64 + (holdDiv-9)
}
```

```
else{
    hex = holdDiv
}
```

divide = uni/16

```
if (divide != 0){
    holdDiv = divide Mod 16
    if (holdDiv > 9){
        hex = 64 + (holdDiv-9)
    }
    else{
        hex = holdDiv
    }
}
```

```
}  
}
```

```
OUTPUT: "The first character of the string is" + chr  
OUTPUT: "It's Unicode value is " + uni + ", binary (" + binary + "), hexadecimal (" + hex + ")"
```

```
uni = (chr2 converted to integer)  
binary = uni Mod 2  
divide = uni/2  
if (divide != 0) {  
    binary = (uni Mod 2) + binary  
    divide = uni/2  
}  
if (divide != 0) {  
    binary = (uni Mod 2) + binary  
    divide = uni/2  
}  
if (divide != 0) {  
    binary = (uni Mod 2) + binary  
    divide = uni/2  
}  
if (divide != 0) {  
    binary = (uni Mod 2) + binary  
    divide = uni/2  
}  
if (divide != 0) {  
    binary = (uni Mod 2) + binary  
    divide = uni/2  
}  
if (divide != 0) {  
    binary = (uni Mod 2) + binary  
    divide = uni/2  
}  
if (divide != 0) {  
    binary = (uni Mod 2) + binary  
    divide = uni/2  
}  
if (divide != 0) {  
    binary = (uni Mod 2) + binary  
    divide = uni/2  
}  
  
int holdDiv = uni Mod 16  
String hex = ""  
if (holdDiv > 9){  
    hex = 64 + (holdDiv-9)  
}  
else{  
    hex = holdDiv  
}  
  
divide = uni/16  
  
if (divide != 0){  
    holdDiv = divide Mod 16  
    if (holdDiv > 9){  
        hex = 64 + (holdDiv-9)  
    }  
    else{  
        hex = holdDiv  
    }  
}
```

OUTPUT: "The last character of the string is" + chr2

OUTPUT: "It's Unicode value is " + uni + ", binary (" + binary + "), hexadecimal (" + hex + ")"

OUTPUT: "App Complete"

```

package labs;
import java.util.Scanner;

public class HLab5 {
    /* ██████████
    * CSCI 1301 T/Th -Marietta
    * Process First and Last Letter
    *
    * Asks user for a single word string and processes first and
    * last letters, converts them to Unicode, binary, and
    * hexadecimal values
    */

    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        System.out.print("Please enter a one word string to process. ");
        String str = input.nextLine();
        char chr = str.charAt(0);
        char chr2 = str.charAt(str.length()-1);

        int uni = chr, divide = 0; String binary = "";
        binary = (uni % 2)+"";
        divide = uni/2;
        if (divide != 0) {binary = (divide % 2) + binary; divide = divide/2;}
        if (divide != 0) {binary = (divide % 2) + binary; divide = divide/2;}
        if (divide != 0) {binary = (divide % 2) + binary; divide = divide/2;}
        if (divide != 0) {binary = (divide % 2) + binary; divide = divide/2;}
        if (divide != 0) {binary = (divide % 2) + binary; divide = divide/2;}
        if (divide != 0) {binary = (divide % 2) + binary; divide = divide/2;}
        if (divide != 0) {binary = (divide % 2) + binary;}

        int holdDiv = uni % 16; String hex = "";
        if (holdDiv > 9) {hex = (char) (64+(holdDiv-9))+"";}
        else {hex = holdDiv+"";}
        divide = uni/16;
        if (divide != 0) {
            holdDiv = divide % 16;
            if (holdDiv > 9) {hex = (char) (64+(holdDiv-9))+hex;}
            else {hex = holdDiv+hex;}
        }

        System.out.println("\nThe first character of the string is "+chr+".");
        System.out.println("Its Unicode value is "+uni+", \nbinary ("+binary
            +"), \nand hexadecimal ("+hex+)");

        uni = chr2; divide = 0; binary = "";
        binary = (uni % 2)+"";
        divide = uni/2;
        if (divide != 0) {binary = (divide % 2) + binary; divide = divide/2;}
        if (divide != 0) {binary = (divide % 2) + binary; divide = divide/2;}
        if (divide != 0) {binary = (divide % 2) + binary; divide = divide/2;}
        if (divide != 0) {binary = (divide % 2) + binary; divide = divide/2;}
        if (divide != 0) {binary = (divide % 2) + binary; divide = divide/2;}
        if (divide != 0) {binary = (divide % 2) + binary; divide = divide/2;}
        if (divide != 0) {binary = (divide % 2) + binary;}

        holdDiv = uni % 16; hex = "";
        if (holdDiv > 9) {hex = (char) (64+(holdDiv-9))+"";}
        else {hex = holdDiv+"";}
        divide = uni/16;
    }
}

```



```
if (divide != 0) {
    holdDiv = divide % 16;
    if (holdDiv > 9) {hex = (char)(64+(holdDiv-9))+hex;}
    else {hex = holdDiv+hex;}
}

System.out.println("\nThe last character of the string is "+chr2+".");
System.out.println("Its Unicode value is "+uni+", \nbinary ("+binary
    +"), \nand hexadecimal("+hex+"");

System.out.println("\n** App Complete **");
input.close();
```

```
}
```

OUTPUT:

Please enter a one word string to process. Testing

The first character of the string is T.
Its Unicode value is 84,
binary (1010100),
and hexadecimal(54)

The last character of the string is g.
Its Unicode value is 103,
binary (1100111),
and hexadecimal(67)

** App Complete **

Please enter a one word string to process. program

The first character of the string is p.
Its Unicode value is 112,
binary (1110000),
and hexadecimal(70)

The last character of the string is m.
Its Unicode value is 109,
binary (1101101),
and hexadecimal(6D)

** App Complete **

Please enter a one word string to process. Application

The first character of the string is A.
Its Unicode value is 65,
binary (1000001),
and hexadecimal(41)

The last character of the string is n.
Its Unicode value is 110,
binary (1101110),
and hexadecimal(6E)

** App Complete **

Please enter a one word string to process. 4thTry

The first character of the string is 4.
Its Unicode value is 52,
binary (110100),
and hexadecimal(34)

The last character of the string is y.
Its Unicode value is 121,

binary (1111001),
and hexadecimal(79)

** App Complete **

Please enter a one word string to process. Thelast

The first character of the string is T.
Its Unicode value is 84,
binary (1010100),
and hexadecimal(54)

The last character of the string is t.
Its Unicode value is 116,
binary (1110100),
and hexadecimal(74)

** App Complete **

Rock-Paper-Scissors-Lizard-Spock

APPLICATION DESCRIPTION:

The user is asked to choose one of the five options (rock, paper, scissor, lizard, or Spock) against the computer's randomly generated option. Whichever player is the first to win three times is the winner of the game.

Rules:

Scissors cut paper

Paper covers rock

Rock smashes lizard

Lizard poisons Spock

Spock breaks scissors

Scissors decapitates lizard

Lizard eats paper

Paper disproves Spock

Spock vaporizes rock

Rock breaks scissors

PSEUDOCODE:

OUTPUT: rules of the game

```
do {
  intPlayer1 = random number between 0-4
  switch (intPlayer1){
    case 0: strPlayer1 = "rock"
    case 1: strPlayer1 = "paper"
    case 2: strPlayer1 = "scissors"
    case 3: strPlayer1 = "lizard"
    default: strPlayer1 = "Spock"
  }
}

do {
  OUTPUT: "Choose one: rock(0), paper(1), scissors(2), lizard(3), Spock(4)"
  INPUT: intPlayer2
  if(!(intPlayer2 > -1 || < 5){OUTPUT: "Invalid selection. Try again."}
}while(!(intPlayer2 > -1 || < 5)
switch (intPlayer2){
  case 0: strPlayer2 = "rock"
  case 1: strPlayer2 = "paper"
  case 2: strPlayer2 = "scissors"
  case 3: strPlayer2 = "lizard"
  default: strPlayer2 = "Spock"
}

OUTPUT: "Computer vs You: "
if (intPlayer1 == intPlayer2){
  OUTPUT: "Both players chose " + strPlayer2
  + ". It's a draw. Keep playing."
}
else if (intPlayer1 == 0){
  if (intPlayer2 == 2){
    OUTPUT: strPlayer1 + " breaks " + strPlayer2 + ". Computer wins"
    player1++
  }
  else if (intPlayer2 == 3){
    OUTPUT: strPlayer1 + " smashes " + strPlayer2 + ". Computer wins"
    player1++
  }
  else if (intPlayer2 == 4){
    OUTPUT: strPlayer1 + " is vaporized by " + strPlayer2 + ". You win!"
    player2++
  }
  else{
    OUTPUT: strPlayer1 + " is covered by " + strPlayer2 + ". You win!"
    player2++
  }
}
else if (intPlayer1 == 1){
  if (intPlayer2 == 2){
    OUTPUT: strPlayer1 + " is cut by " + strPlayer2 + ". You win!"
    player2++
  }
  else if (intPlayer2 == 3){
    OUTPUT: strPlayer1 + " is eaten by " + strPlayer2 + ". You win!"
    player2++
  }
  else if (intPlayer2 == 4){
    OUTPUT: strPlayer1 + " disproves " + strPlayer2 + ". Computer wins"
```

```

    player1++
}
else{
    OUTPUT: strPlayer1 + " covers " + strPlayer2 + ". Computer wins"
    player1++
}
}
else if (intPlayer1 == 2){
    if (intPlayer2 == 0){
        OUTPUT: strPlayer1 + " are broken by " + strPlayer2 + ". You win!"
        player2++
    }
    else if (intPlayer2 == 3){
        OUTPUT: strPlayer1 + " decapitate " + strPlayer2 + ". Computer wins"
        player1++
    }
    else if (intPlayer2 == 4){
        OUTPUT: strPlayer1 + " are broken by " + strPlayer2 + ". You win!"
        player2++
    }
    else{
        OUTPUT: strPlayer1 + " cut " + strPlayer2 + ". Computer wins"
        player1++
    }
}
else if (intPlayer1 == 3){
    if (intPlayer2 == 0){
        OUTPUT: strPlayer1 + " is smashed by " + strPlayer2 + ". You win!"
        player2++
    }
    else if (intPlayer2 == 2){
        OUTPUT: strPlayer1 + " is decapitated by " + strPlayer2 + ". You win!"
        player2++
    }
    else if (intPlayer2 == 4){
        OUTPUT: strPlayer1 + " poisons " + strPlayer2 + ". Computer wins"
        player1++
    }
    else{
        OUTPUT: strPlayer1 + " eats " + strPlayer2 + ". Computer wins"
        player1++
    }
}
else {
    if (intPlayer2 == 0){
        OUTPUT: strPlayer1 + " vaporizes " + strPlayer2 + ". Computer wins"
        player1++
    }
    else if (intPlayer2 == 2){
        OUTPUT: strPlayer1 + " breaks " + strPlayer2 + ". Computer wins"
        player1++
    }
    else if (intPlayer2 == 3){
        OUTPUT: strPlayer1 + " is poisoned by " + strPlayer2 + ". You win!"
        player2++
    }
    else{
        OUTPUT: strPlayer1 + " is disproved by " + strPlayer2 + ". You win!"
        player2++
    }
}
}

```

```
}  
}while(player1 && player2 < 3)  
  
if (player1 > player2){OUTPUT: "The computer won the game.}  
else {OUTPUT: "You won the game!"}
```

```

package labs;
import java.util.Scanner;

public class HLab6 {
    /*
     * CSC1 1301 T/Th -Marietta
     * Rock-Paper-Scissors-Lizard-Spock
     *
     * Plays the game "Rock-Paper-Scissors-Lizard-Spock"
     * until one player wins 3
     */

    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        int intPlayer1=0, intPlayer2=0;
        String strPlayer1="", strPlayer2="";
        int player1=0, player2=0;
        System.out.println("Rules:");
        System.out.println(" Scissors cut paper");
        System.out.println(" Paper covers rock");
        System.out.println(" Rock smashes lizard");
        System.out.println(" Lizard poisons Spock");
        System.out.println(" Spock breaks scissors");
        System.out.println(" Scissors decapitates lizard");
        System.out.println(" Lizard eats paper");
        System.out.println(" Paper disproves Spock");
        System.out.println(" Spock vaporizes rock");
        System.out.println(" Rock breaks scissors\n");

        do {
            // Determine computer choice
            intPlayer1 = ((int) (Math.random()*10)/2);
            switch(intPlayer1) {
                case 0: strPlayer1 = "rock"; break;
                case 1: strPlayer1 = "paper"; break;
                case 2: strPlayer1 = "scissors"; break;
                case 3: strPlayer1 = "lizard"; break;
                default: strPlayer1 = "Spock";
            }

            // Determine person's choice & check allowed
            do {
                System.out.print(
                    "Choose one: rock(0), paper(1), scissors(2),"
                    + " lizard(3), Spock(4) ");
                intPlayer2 = input.nextInt();
                if (!(intPlayer2>-1 && intPlayer2<5)) {
                    System.out.println("Invalid selection. Try again.\n");
                }
            }while(!(intPlayer2>-1 && intPlayer2<5));
            switch(intPlayer2) {
                case 0: strPlayer2 = "rock"; break;
                case 1: strPlayer2 = "paper"; break;
                case 2: strPlayer2 = "scissors"; break;
                case 3: strPlayer2 = "lizard"; break;
                default: strPlayer2 = "Spock";
            }

            // Determines who wins round and output given
            String printOut = "";

```



```

if (intPlayer1 == intPlayer2) {
    printOut = "Both players chose "+strPlayer2
        +". It's a draw. Keep playing.";
}
else if(intPlayer1==0) {
    if (intPlayer2==2) { // rock vs scissors, rock wins
        printOut = strPlayer1+" breaks "+strPlayer2+". Computer wins";
        player1++;
    }
    else if (intPlayer2==3) { // rock vs lizard, rock wins
        printOut = strPlayer1+" smashes "+strPlayer2+". Computer wins";
        player1++;
    }
    else if (intPlayer2==4) { // rock vs Spock, Spock wins
        printOut = strPlayer1+" is vaporized by "+strPlayer2+". You win!";
        player2++;
    }
    else { // rock vs paper, paper wins
        printOut = strPlayer1+" is covered by "+strPlayer2+". You win!";
        player2++;
    }
}
else if (intPlayer1==1) {
    if (intPlayer2==2) { // paper vs scissors, scissors wins
        printOut = strPlayer1+" is cut by "+strPlayer2+". You win!";
        player2++;
    }
    else if (intPlayer2==3) { // paper vs lizard, lizard wins
        printOut = strPlayer1+" is eaten by "+strPlayer2+". You win!";
        player2++;
    }
    else if (intPlayer2==4) { // paper vs Spock, paper wins
        printOut = strPlayer1+" disproves "+strPlayer2+". Computer wins";
        player1++;
    }
    else { // paper vs rock, paper wins
        printOut = strPlayer1+" covers "+strPlayer2+". Computer wins";
        player1++;
    }
}
else if (intPlayer1==2) {
    if (intPlayer2==0) { // scissors vs rock, rock wins
        printOut = strPlayer1+" are broken by "+strPlayer2+". You win!";
        player2+=1;
    }
    else if (intPlayer2==3) { // scissors vs lizard, scissors wins
        printOut = strPlayer1+" decapitate "+strPlayer2+". Computer wins";
        player1+=1;
    }
    else if (intPlayer2==4) { // scissors vs Spock, Spock wins
        printOut = strPlayer1+" are broken by "+strPlayer2+". You win!";
        player2+=1;
    }
    else { // scissors vs paper, scissors wins
        printOut = strPlayer1+" cut "+strPlayer2+". Computer wins";
        player1+=1;
    }
}
else if (intPlayer1==3) {
    if (intPlayer2==0) { // lizard vs rock, rock wins

```

```

        printOut = strPlayer1+" is smashed by "+strPlayer2+". You win!";
        player2+=1;
    }
    else if (intPlayer2==2) { // lizard vs scissors, scissors win
        printOut = strPlayer1+" is decapitated by "+strPlayer2
            +". You win!";
        player2+=1;
    }
    else if (intPlayer2==4) { // lizard vs Spock, lizard wins
        printOut = strPlayer1+" poisons "+strPlayer2+". Computer wins";
        player1+=1;
    }
    else { // lizard vs paper, lizard wins
        printOut = strPlayer1+" eats "+strPlayer2+". Computer wins";
        player1+=1;
    }
}
else {
    if (intPlayer2==0) { // Spock vs rock, Spock wins
        printOut = strPlayer1+" vaporizes "+strPlayer2+". Computer wins";
        player1+=1;
    }
    else if (intPlayer2==2) { // Spock vs scissors, Spock wins
        printOut = strPlayer1+" breaks "+strPlayer2+". Computer wins";
        player1+=1;
    }
    else if (intPlayer2==3) { // Spock vs lizard, lizard wins
        printOut = strPlayer1+" is poisoned by "+strPlayer2+". You win!";
        player2+=1;
    }
    else { // Spock vs paper, paper wins
        printOut = strPlayer1+" is dispoved by "+strPlayer2+". You win!";
        player2+=1;
    }
}
    System.out.println("Computer vs You: "+printOut+"\n");
}while(player1<3 && player2<3);

// Determines who wins game and output for it
if (player1>player2) {System.out.println("\nThe computer won the game.");}
else {System.out.println("\nYou won the game!");}
System.out.println("\n** App Complete **");

input.close(); }

```

OUTPUT:

Rules:

Scissors cut paper
Paper covers rock
Rock smashes lizard
Lizard poisons Spock
Spock breaks scissors
Scissors decapitates lizard
Lizard eats paper
Paper disproves Spock
Spock vaporizes rock
Rock breaks scissors

Choose one: rock(0), paper(1), scissors(2), lizard(3), Spock(4) 0
Computer vs You: Both players chose rock. It's a draw. Keep playing.

Choose one: rock(0), paper(1), scissors(2), lizard(3), Spock(4) 0
Computer vs You: lizard is smashed by rock. You win!

Choose one: rock(0), paper(1), scissors(2), lizard(3), Spock(4) 0
Computer vs You: Both players chose rock. It's a draw. Keep playing.

Choose one: rock(0), paper(1), scissors(2), lizard(3), Spock(4) 0
Computer vs You: Spock vaporizes rock. Computer wins

Choose one: rock(0), paper(1), scissors(2), lizard(3), Spock(4) 0
Computer vs You: lizard is smashed by rock. You win!

Choose one: rock(0), paper(1), scissors(2), lizard(3), Spock(4) 0
Computer vs You: lizard is smashed by rock. You win!

You won the game!

** App Complete **

Rules:

Scissors cut paper
Paper covers rock
Rock smashes lizard
Lizard poisons Spock
Spock breaks scissors
Scissors decapitates lizard
Lizard eats paper
Paper disproves Spock
Spock vaporizes rock
Rock breaks scissors

Choose one: rock(0), paper(1), scissors(2), lizard(3), Spock(4) 1
Computer vs You: Both players chose paper. It's a draw. Keep playing.

Choose one: rock(0), paper(1), scissors(2), lizard(3), Spock(4) 1
Computer vs You: rock is covered by paper. You win!

Choose one: rock(0), paper(1), scissors(2), lizard(3), Spock(4) 1
Computer vs You: rock is covered by paper. You win!

Choose one: rock(0), paper(1), scissors(2), lizard(3), Spock(4) 1
Computer vs You: scissors cut paper. Computer wins

Choose one: rock(0), paper(1), scissors(2), lizard(3), Spock(4) 1
Computer vs You: scissors cut paper. Computer wins

Choose one: rock(0), paper(1), scissors(2), lizard(3), Spock(4) 1
Computer vs You: Both players chose paper. It's a draw. Keep playing.

Choose one: rock(0), paper(1), scissors(2), lizard(3), Spock(4) 1
Computer vs You: rock is covered by paper. You win!

You won the game!

** App Complete **

Rules:

Scissors cut paper
Paper covers rock
Rock smashes lizard
Lizard poisons Spock
Spock breaks scissors
Scissors decapitates lizard
Lizard eats paper
Paper disproves Spock
Spock vaporizes rock
Rock breaks scissors

Choose one: rock(0), paper(1), scissors(2), lizard(3), Spock(4) 2
Computer vs You: Spock breaks scissors. Computer wins

Choose one: rock(0), paper(1), scissors(2), lizard(3), Spock(4) 2
Computer vs You: Both players chose scissors. It's a draw. Keep playing.

Choose one: rock(0), paper(1), scissors(2), lizard(3), Spock(4) 2
Computer vs You: lizard is decapitated by scissors. You win!

Choose one: rock(0), paper(1), scissors(2), lizard(3), Spock(4) 2
Computer vs You: lizard is decapitated by scissors. You win!

Choose one: rock(0), paper(1), scissors(2), lizard(3), Spock(4) 2
Computer vs You: lizard is decapitated by scissors. You win!

You won the game!

** App Complete **

Rules:

Scissors cut paper
Paper covers rock
Rock smashes lizard
Lizard poisons Spock
Spock breaks scissors

Scissors decapitates lizard
Lizard eats paper
Paper disproves Spock
Spock vaporizes rock
Rock breaks scissors

Choose one: rock(0), paper(1), scissors(2), lizard(3), Spock(4) 3
Computer vs You: rock smashes lizard. Computer wins

Choose one: rock(0), paper(1), scissors(2), lizard(3), Spock(4) 3
Computer vs You: Both players chose lizard. It's a draw. Keep playing.

Choose one: rock(0), paper(1), scissors(2), lizard(3), Spock(4) 3
Computer vs You: rock smashes lizard. Computer wins

Choose one: rock(0), paper(1), scissors(2), lizard(3), Spock(4) 3
Computer vs You: rock smashes lizard. Computer wins

The computer won the game.

** App Complete **

Rules:

Scissors cut paper
Paper covers rock
Rock smashes lizard
Lizard poisons Spock
Spock breaks scissors
Scissors decapitates lizard
Lizard eats paper
Paper disproves Spock
Spock vaporizes rock
Rock breaks scissors

Choose one: rock(0), paper(1), scissors(2), lizard(3), Spock(4) 4
Computer vs You: scissors are broken by Spock. You win!

Choose one: rock(0), paper(1), scissors(2), lizard(3), Spock(4) 4
Computer vs You: rock is vaporized by Spock. You win!

Choose one: rock(0), paper(1), scissors(2), lizard(3), Spock(4) 4
Computer vs You: rock is vaporized by Spock. You win!

You won the game!

** App Complete **

Rules:

Scissors cut paper
Paper covers rock
Rock smashes lizard
Lizard poisons Spock
Spock breaks scissors

Scissors decapitates lizard
Lizard eats paper
Paper disproves Spock
Spock vaporizes rock
Rock breaks scissors

Choose one: rock(0), paper(1), scissors(2), lizard(3), Spock(4) 5
Invalid selection. Try again.

Choose one: rock(0), paper(1), scissors(2), lizard(3), Spock(4) -1
Invalid selection. Try again.

Choose one: rock(0), paper(1), scissors(2), lizard(3), Spock(4) 0
Computer vs You: Spock vaporizes rock. Computer wins

Choose one: rock(0), paper(1), scissors(2), lizard(3), Spock(4) 1
Computer vs You: Spock is disproved by paper. You win!

Choose one: rock(0), paper(1), scissors(2), lizard(3), Spock(4) 2
Computer vs You: Both players chose scissors. It's a draw. Keep playing.

Choose one: rock(0), paper(1), scissors(2), lizard(3), Spock(4) 3
Computer vs You: Both players chose lizard. It's a draw. Keep playing.

Choose one: rock(0), paper(1), scissors(2), lizard(3), Spock(4) 4
Computer vs You: paper disproves Spock. Computer wins

Choose one: rock(0), paper(1), scissors(2), lizard(3), Spock(4) 0
Computer vs You: Both players chose rock. It's a draw. Keep playing.

Choose one: rock(0), paper(1), scissors(2), lizard(3), Spock(4) 1
Computer vs You: rock is covered by paper. You win!

Choose one: rock(0), paper(1), scissors(2), lizard(3), Spock(4) 2
Computer vs You: paper is cut by scissors. You win!

You won the game!

** App Complete **

Temperature

APPLICATION DESCRIPTION:

The application converts a range of temperatures from Celsius to Fahrenheit and from Fahrenheit to Celsius. The conversion processes are accessed using methods. The output is given in a clear formatted table.

Temperature (pseudocode)

```
(METHOD: main)
celsius = 40
fahrenheit = 120
OUTPUT: formatted into columns "Celsius", "Fahrenheit", "|", "Fahrenheit",
"Celsius"
while(celsius>30){
  OUTPUT: ""
  OUTPUT: formatted into columns
    celsius, celsiusToFahrenheit(celsius), "|", fahrenheit,
    fahrenheitToCelsius(fahrenheit)
  celsius--
  fahrenheit -= 10
}

(METHOD: celsiusToFahrenheit)
fahrenheit = (9.0/5)* celsius + 32
return fahrenheit

(METHOD: fahrenheitToCelsius)
celsius = (5.0/9)* (fahrenheit - 32)
return celsius
```



```
package labs;

public class Lab7_1 {
    /* ██████████
    * CSCI 1301 T/Th -Marietta
    * Temperature
    *
    * Convert Fahrenheit to Celsius and
    * Celsius to Fahrenheit using methods
    */

    public static void main(String[] args) {
        double celsius = 40, fahrenheit = 120;
        System.out.printf("%-10s %-14s %-4s %-14s %s", "Celsius", "Fahrenheit",
            "|", "Fahrenheit", "Celsius");
        while (celsius > 30) {
            System.out.println();
            System.out.printf("%-10.1f %-14.1f %-4s %-14.1f %.2f",
                celsius, celsiusToFahrenheit(celsius), "|", fahrenheit,
                fahrenheitToCelsius(fahrenheit));
            celsius--; fahrenheit -= 10;
        }
    }

    public static double celsiusToFahrenheit(double celsius) {
        double fahrenheit = (9.0 / 5) * celsius + 32;
        return fahrenheit;
    }

    public static double fahrenheitToCelsius(double fahrenheit) {
        double celsius = (5.0 / 9) * (fahrenheit - 32);
        return celsius;
    }
}
```

OUTPUT:

Celsius	Fahrenheit		Fahrenheit	Celsius
40.0	104.0		120.0	48.89
39.0	102.2		110.0	43.33
38.0	100.4		100.0	37.78
37.0	98.6		90.0	32.22
36.0	96.8		80.0	26.67
35.0	95.0		70.0	21.11
34.0	93.2		60.0	15.56
33.0	91.4		50.0	10.00
32.0	89.6		40.0	4.44
31.0	87.8		30.0	-1.11

Highest Scoring Student

APPLICATION DESCRIPTION:

The user inputs any number of students along with their corresponding score, and the application will return the student with the highest score. The user is asked to enter the name and score of each student. To end the program and get the highest score, the user must enter "alldone".

Highest Scoring Student (psuedocode)

Create variables: name, student, number, and score
Initialize student with empty string
Initialize number and score with 0

Create and initialize Scanner object
Output instructions including which exit word to use
Output request for name
Input name

Loop until input is exit word
 Output request for score
 Input number
 if number is > score then
 score and studen = number and name
 Output request for name
 Input name
end loop

Output student with the highest grade
close Scanner object

```

package labs;
import java.util.Scanner;

public class Lab8 {
    /*
    * CSC1 1301 T/Th -Marietta
    * Highest Scoring Student
    *
    * Accepts unlimited number of students and their exam score and returns
    * student with highest score.
    */

    public static void main(String[] args) {
        String name, student="";
        int number=0, score=0;
        Scanner input = new Scanner (System.in);
        System.out.println(" **Instructions** to exit early, enter: alldone\n");
        System.out.print("What is the student's name? ");
        name = input.next();
        while (!name.equals("alldone")) {
            System.out.print("What is the student's score? ");
            number = input.nextInt();
            if (isGreaterThan(number,score)) {score = number; student = name;}
            System.out.print("What is the student's name? ");
            name = input.next();
        }
        System.out.println(
            "\n" + student + " had the highest grade, with a score of " + score);
        input.close();
    }

    public static boolean isGreaterThan(int score1, int score2) {
        if (score1>score2) {return true;}
        else {return false;}
    }
}

```

Output:

Instructions to exit early, enter: alldone

What is the student's name? jack
What is the student's score? 70
What is the student's name? sam
What is the student's score? 80
What is the student's name? R2D2
What is the student's score? 90
What is the student's name? alldone

R2D2 had the highest grade, with a score of 90

Instructions to exit early, enter: alldone

What is the student's name? jack
What is the student's score? 70
What is the student's name? R2D2
What is the student's score? 90
What is the student's name? sam
What is the student's score? 80
What is the student's name? alldone

R2D2had the highest grade, with a score of 90

Instructions to exit early, enter: alldone

What is the student's name? sam
What is the student's score? 80
What is the student's name? jack
What is the student's score? 70
What is the student's name? R2D2
What is the student's score? 90
What is the student's name? alldone

R2D2 had the highest grade, with a score of 90

Instructions to exit early, enter: alldone

What is the student's name? sam
What is the student's score? 80
What is the student's name? R2D2
What is the student's score? 90
What is the student's name? jack
What is the student's score? 70
What is the student's name? alldone

R2D2 had the highest grade, with a score of 90

Instructions to exit early, enter: alldone

What is the student's name? R2D2
What is the student's score? 90
What is the student's name? jack
What is the student's score? 70
What is the student's name? sam
What is the student's score? 80
What is the student's name? alldone

R2D2 had the highest grade, with a score of 90

Instructions to exit early, enter: alldone

What is the student's name? R2D2
What is the student's score? 90
What is the student's name? sam
What is the student's score? 80
What is the student's name? jack
What is the student's score? 70
What is the student's name? alldone

R2D2 had the highest grade, with a score of 90

Calculate Restaurant Check

APPLICATION DESCRIPTION:

The program accepts the customers 3 chosen items on the menu as well as any discounts applicable. Once all information is entered, it returns each customer's subtotal as well as the grand total for that check. The food and discount menus are provided for reference.

Calculate Restaurant Check (psuedocode)

Create and initialize global variable Scanner object

Create and initialize global arrays: itemName and ItemCost

Create variables: numOfCost, code, item, total, custTotal, custSubTotal

Initialize numOfCost, code, and item to -1

Initialize total and custSubTotal to 0

Loop until numOfCost is not less than 0

 Output request for number of customers

 Input numOfCust

 If numOfCust = 0 then Output exiting app notice

 Else if numOfCust < 1 then Output notice that there are not enough customers
end loop

If numOfCust >0

 Loop through number of customers

 reset custTotal to 0

 Output which is the current customer working with

 Output discount code menu and request for choice

 Loop until code != -1

 Input code

 If code < 1 or > 4 then Output notice invalid and reset code to -1
 end loop

 Output food menu

 Loop through 3 items

 Loop until item is between 0 and 8 (exclude 0 and 8)

 Output request for chosen item

 Input item

 if item < 0 or > 8 then Output notice invalid

 Switch code

 if 1 then custTotal + child subtotal and discount

 if 2 or 3 then custTotal + teen/senior subtotal and discount

 if 4 then custTotal + subtotal and no discount

 reset item to -1

 end loop

 If code = 4 then custTotal = custTotal * 1.05

 custTotal = round-up custTotal*100

 custTotal = custTotal / 100

 Output current customer total

 total = total + custTotal

 reset code to -1

 end loop

 Output total

```

package labs;
import java.util.Scanner;

public class Lab9 {
    /* ██████████
    * CSCI 1301 T/Th -Marietta
    * Calculate Restaurant Check
    *
    * Calculates the cost of a party's order at a restaurant
    */

    public static Scanner input = new Scanner (System.in);
    public static String[] itemName =
        {"Soup", "Wings", "Burger", "Chicken sandwich",
        "Fries", "Pie", "Ice cream", "Soft drink", "coffee"};
    public static double[] itemCost =
        {2.50, .15, 4.95, 5.95, 1.99, 2.95, 2.99, 1.50, 1.00};

    public static void main(String[] args) {
        // tracks # of cust, discount code, current menu item
        int numOfCust = -1, code = -1, item = -1;
        // tracks grand total, customer total, customer working total
        double total = 0, custTotal, custSubTotal = 0;

        while(numOfCust < 0) {
            System.out.print("How many people in your party? ");
            numOfCust = input.nextInt();
            if (numOfCust == 0) {System.out.println("Exiting App");}
            else if (numOfCust < 1) {
                System.out.println("Not enough customers. Please try again");
            }
        }
        if (numOfCust > 0) {
            for (int c = 0; c < numOfCust; c++) {
                custTotal = 0; // reset
                System.out.println(
                    "\nOrder information for customer " + (c+1) + "... \n");
                System.out.println(
                    "Please enter the code (1-4) for any applicable discounts");
                System.out.print("1 - if CHILD 5 years old or younger\n"
                    + "2 - if TEEN between 13 and 19\n"
                    + "3 - if SENIOR 65 years old or older\n"
                    + "4 - if NONE of the above\n");
                while (code == -1) {
                    code = input.nextInt();
                    if (code < 1 || code > 4) {
                        System.out.println("Invalid code. Please try again.");
                        code = -1;
                    }
                }
            }
            System.out.println("\nItems on the menu:");
            System.out.print("1 " + itemName[0] + "\n"
                + "2 " + itemName[1] + "\n"
                + "3 " + itemName[2] + "\n"
                + "4 " + itemName[3] + "\n"
                + "5 " + itemName[4] + "\n"
                + "6 " + itemName[5] + "\n"
                + "7 " + itemName[6] + "\n"
                + "8 " + itemName[7] + "\n");
        }
    }
}

```

```

    + "9 " + itemName[8] + "\n\n");
for (int i = 0; i < 3; i++) {
    while (item < 0 || item > 8) {
        System.out.print("Please type in item " + (i+1) + " (1-9) ");
        item = input.nextInt()-1;
        if (item < 0 || item > 8) {
            System.out.println("Invalid entry. Please try again.");
        }
    }
    switch(code) {
        case 1: custTotal += child(item); break;
        case 2:
        case 3: custTotal += teenSenior(item); break;
        case 4: custSubTotal = regular(item);
            custTotal += custSubTotal; break;
    }
    item = -1; // reset
}
if (code == 4) {custTotal = custTotal * 1.05;}
custTotal = Math.round(custTotal * 100);
custTotal /= 100;
System.out.printf(
    "The total for customer 1 is $%.2f (discount code: "
    + code + ")\n", custTotal);
total += custTotal;
code = -1; // reset
}
System.out.println("\n\nGrand Total for "
    + numOfCust + " people: " + total);
}

System.out.println("\n** App Complete **");
input.close();
}

public static double child (int item) {double x = regular(item); return 0.0;}
public static double teenSenior(int item) {
    double cost = regular(item);
    return cost - (cost * 0.25);
}
public static double regular(int item) {
    int multiply = 1;
    if (item == 1) {
        System.out.print("How many wings does the customer want? ");
        multiply = input.nextInt();
    }
    return itemCost[item] * multiply;
}
}
}

```

Output:

How many people in your party? 4

Order information for customer 1...

Please enter the code (1-4) for any applicable discounts

- 1 - if CHILD 5 years old or younger
 - 2 - if TEEN between 13 and 19
 - 3 - if SENIOR 65 years old or older
 - 4 - if NONE of the above
- 4

Items on the menu:

- 1 Soup
- 2 Wings
- 3 Burger
- 4 Chicken sandwich
- 5 Fries
- 6 Pie
- 7 Ice cream
- 8 Soft drink
- 9 coffee

Please type in item 1 (1-9) 3

Please type in item 2 (1-9) 2

How many wings does the customer want? 5

Please type in item 3 (1-9) 9

The total for customer 1 is \$7.04 (discount code: 4)

Order information for customer 2...

Please enter the code (1-4) for any applicable discounts

- 1 - if CHILD 5 years old or younger
 - 2 - if TEEN between 13 and 19
 - 3 - if SENIOR 65 years old or older
 - 4 - if NONE of the above
- 4

Items on the menu:

- 1 Soup
- 2 Wings
- 3 Burger
- 4 Chicken sandwich
- 5 Fries
- 6 Pie
- 7 Ice cream
- 8 Soft drink
- 9 coffee

Please type in item 1 (1-9) 3

Please type in item 2 (1-9) 1

Please type in item 3 (1-9) 8

The total for customer 1 is \$9.40 (discount code: 4)

Order information for customer 3...

Please enter the code (1-4) for any applicable discounts

- 1 - if CHILD 5 years old or younger
 - 2 - if TEEN between 13 and 19
 - 3 - if SENIOR 65 years old or older
 - 4 - if NONE of the above
- 3

Items on the menu:

- 1 Soup
- 2 Wings
- 3 Burger
- 4 Chicken sandwich
- 5 Fries
- 6 Pie
- 7 Ice cream
- 8 Soft drink
- 9 coffee

Please type in item 1 (1-9) 3
Please type in item 2 (1-9) 1
Please type in item 3 (1-9) 9
The total for customer 1 is \$6.34 (discount code: 3)

Order information for customer 4...

Please enter the code (1-4) for any applicable discounts

- 1 - if CHILD 5 years old or younger
 - 2 - if TEEN between 13 and 19
 - 3 - if SENIOR 65 years old or older
 - 4 - if NONE of the above
- 1

Items on the menu:

- 1 Soup
- 2 Wings
- 3 Burger
- 4 Chicken sandwich
- 5 Fries
- 6 Pie
- 7 Ice cream
- 8 Soft drink
- 9 coffee

Please type in item 1 (1-9) 3
Please type in item 2 (1-9) 8
Please type in item 3 (1-9) 7
The total for customer 1 is \$0.00 (discount code: 1)

Grand Total for 4 people: 22.78

** App Complete **