Computer Science:

Project and Practical Record

Sarang, XI

Year 19-20

Contents

Project: Hangman

Code:

```
import java.util.Arrays; import java.util.Scanner;
public class Hangman {

  //Displays message when letters are repeated. To reduce code.
  public static void reguess(String u,StringBuffer ub, char x, char[] b,int c ){
    System.out.println(x + " has already been guessed. Guess another letter.");
    System.out.println((Arrays.toString(b)) + "          Lives remaining: " + c);
    System.out.println("                        Used = "+u);}

  //Main function of the game
  public static void play(String word, int chances) {
    char[] board = new char[chances];int i;
    for (i = 0; i < chances; i++) {    //Displays Game progress
      board[i] = '-';}
    System.out.println((Arrays.toString(board)) + "          Lives remaining: " + chances);
    Scanner inp = new Scanner(System.in);
    StringBuffer used = new StringBuffer(); String used_str = String.valueOf(used);
    while (chances > 0) {  //enables Multiple attempts
      System.out.print("Guess a letter: ");   //stores user's guess
      char x = inp.next().charAt(0);
      used_str = String.valueOf(used);
      if (word.contains(x + "")) {          //If guess is right...
        if (used_str.contains(x + "")) {    //If already guessed...
          reguess(used_str,used,x,board,chances); //...guess again
          continue;}
        else {  System.out.println("[GOOD GUESS!]");    //Correct guess
            for (i = 0; i < chances; i++) {    //updates progress
              if (word.charAt(i) == x) {
                board[i] = x;}}
            used.append(x);}}
      else {  if (used_str.contains(x + "")) {    //If wrong guess...
          reguess(used_str,used,x,board,chances);   //...and repeated
          continue;}
        else {   System.out.println("[WRONG GUESS!]");   //chance lost
            chances--;
            used.append(x);}}
```

```
        if (word.equalsIgnoreCase(String.valueOf(board))) {    //If player wins
           System.out.println("You won with " + chances + " guesses remaining! the word was " +
word);
           break;}
        else {  System.out.println((Arrays.toString(board)) + "          Lives remaining: " + chances);
              System.out.println("                        Used = "+used);}}
      if(chances==0){       //If player loses
        System.out.println("Better luck next time :(");
        System.out.println("The word was " + word);}}

   public static void main(String[] args) {
      Scanner input = new Scanner(System.in);
      System.out.print("Enter secretword: ");
      String secret = input.next();   //Stores user's secretword
      play(secret, secret.length());}}    //Starts the functioning of the game
```

Trial:

```
Enter secretword: java
[-, -, -, -]          Lives remaining: 4
Guess a letter: s
[WRONG GUESS!]
[-, -, -, -]          Lives remaining: 3
                        Used = s
Guess a letter: a
[GOOD GUESS!]
[-, a, -, a]          Lives remaining: 3
                        Used = sa
Guess a letter: v
[GOOD GUESS!]
[-, a, v, a]          Lives remaining: 3
                        Used = sav
Guess a letter: s
s has already been guessed. Guess another letter.
[-, a, v, a]          Lives remaining: 3
                        Used = sav
Guess a letter: j
[GOOD GUESS!]
You won with 3 guesses remaining! the word was java
```

Practical Record:

1. Fibonacci

Code:

```
import java.util.Scanner;
public class Fibonacci {
   public static void main(String[] args) {
      Scanner input = new Scanner(System.in);
      System.out.print("Enter length (<92) of Fibonacci series required: ");
      int n = input.nextInt(); //stores length of fib from user
      long a = 1; long f = 0; //Initialization...
      System.out.println("Fibonacci Series:");
      System.out.println("0");
      for(int i=1;i<n;i++) { //Generates fibonacci sequence
         a = f+a; f = a-f;
         System.out.println(f);}}}
```

Trial:

```
Enter length (<92) of Fibonacci series required: 6
Fibonacci Series:
0
1
1
2
3
5
```

---

2. PrimeCheck

Code:

```
import java.util.Scanner;
public class PrimeCheck {
   public static void main(String[] args) {
      Scanner input = new Scanner(System.in);
      System.out.print("Enter a number: ");
      int n = input.nextInt(); //receives input
      int x = 0;
      for (int i = 1; i<=n; i++) {
         if(n%i==0) { //checks divisibility
```

```
        x+=1;}}
    if(x==2) {      //decides output
       System.out.println(n+" is Prime");}
    else {  System.out.println(n+" is not Prime");}}}
```

Trial:

Enter a number: 2
2 is Prime

Enter a number: 27
27 is not prime

---

3. SumDigits

Code:

```
import java.util.Scanner;
public class SumDigits {

   public static long sumDigits(long n){    //sums the digits
      long sum = 0;
      while(n>0){
         long digit = n%10; sum+=digit; n/=10;}
      return sum;}

   public static void main(String[] args) {
      Scanner input = new Scanner(System.in);
      System.out.print("Enter a number: ");
      long x = input.nextLong();
      System.out.println(sumDigits(x));}}
```

Trial:

Enter a number: 1908
18

Enter a number: 100
1

---

4. LeastNumberSum

Code:

```
import java.util.Scanner;
public class LeastNumberSum {

   public static int sumDigits(int n){  //sums the digit of M
      int sum = 0; int r;  //sum holds value for sum of digits while r stores remainder
      while(n>0){
```

```
        r=n%10; sum+=r; n/=10;}
    return sum;}

  public static void main(String[] args) {
    int N;int M; // Positive integers M & N values are stored.
    Scanner input = new Scanner(System.in);
    System.out.print("Enter N(0<N<100): ");
    N = input.nextInt();
    System.out.print("Enter M(100<=M<10000): ");
    M = input.nextInt();
    if((N>100)||(N<1)){
      System.out.print("Invalid input");}
    else if((M>9999)||(M<100)){
      System.out.print("Invalid input");}
    else{          // boolean check checks whether sum a given number M equals N
      boolean check = true; int a; int count=0; //count stores length of number
      while(check){
      a=M;
      if(sumDigits(a)==N){
        System.out.println("The required number = "+M);
        check=false;}
      else{ M+=1;}}
    while(M>0){
      M/=10; count+=1;}
    System.out.println("Total number of digits = "+count);}}}
```

Trial:

```
Enter N(0<N<100): 101
Enter M(100<=M<10000): 70
Invalid input
```

```
Enter N(0<N<100): 35
Enter M(100<=M<10000): 898
The required number = 8999
Total number of digits = 4
```

---

5. MultiplicationTables

Code:

```
import java.util.Scanner;
public class MultiplicationTables {
  public static void main(String[] args) {
    Scanner inp = new Scanner(System.in);
    System.out.print("Enter a number: ");
    int n = inp.nextInt();
```

```
    for(int i=1;i<=10;i++) {            //Receives a number and prints its table.
      System.out.println(n+" x "+i+" = "+(i*n));}}}
```

Trial:

```
Enter a number: 17
17 x 1 = 17
17 x 2 = 34
17 x 3 = 51
17 x 4 = 68
17 x 5 = 85
17 x 6 = 102
17 x 7 = 119
17 x 8 = 136
17 x 9 = 153
17 x 10 = 170
```

---

6. WordCount

Code:

```
import java.util.Scanner;
public class WordCount {
   public static void main(String[] args) {
      Scanner input = new Scanner(System.in);
      System.out.print("Enter text: ");
      String x = input.nextLine();
      String [] words = x.split(" ");          //recognizes words by blanks and counts
      System.out.println(words.length);}}
```

Trial:

```
Enter text: This is sample input          Enter text: let us take a bit of it.
4                                         7
```

---

7. UserInputValidation

Code:

```
import java.util.*;
public class UserInputValidation {
   public static void main(String[] args) {
      Scanner input = new Scanner(System.in);
      int var1,var2; boolean valid = false;
```

```
    do{ try{                        //if any input throws an error,
          System.out.print("Enter var1: "); //all other inputs must be re-entered.
          var1 = input.nextInt();
          System.out.print("Enter var2: ");
          var2 = input.nextInt();
          //...+all other user interactions
          valid=true;}  //must  be last line
       catch (InputMismatchException e){
          System.out.print("Please enter valid input.  ");
          System.err.println(e);
          input.nextLine();}
    } while(!valid);}}
```

Trial:

```
Enter var1: abcd
Please enter valid input.  java.util.InputMismatchException
Enter var1: 2
Enter var2: 3
```

---

8. PalindromeCheck

Code:

```
import java.util.Scanner;
public class PalindromeCheck {
   public static void main(String[] args) {
      Scanner input = new Scanner(System.in);
      System.out.print("Type any text: ");
      String s = input.next();
      int l = s.length();          //checks whether word is symmetrical
      boolean b = true;
      for(int i=0;i<l/2;i++) {
         if(s.charAt(i)!= s.charAt(l-1-i)) {
            b = false; break;}}
      if(b==false){
         System.out.println(s+" is not a palindrome");}
      else{   System.out.println(s+" is a palindrome");}}}
```

Trial:

```
1) Type any word: malayalam
   malayalam is a palindrome

2) Type any word: mathematics
   mathematics is not a palindrome
```

---

9. PalindromeCheck_recursive

Code:

```
import java.util.Scanner;
public class PalindromeCheck_recursive {
   public static boolean PalCheck(String s){
      int l = s.length();                          //uses recursion to check
      if(l<=1){ return true;}                      //whether word is palidrome
      else{  if(s.charAt(0)==s.charAt(l-1)){
               return PalCheck(s.substring(1,l-1));}
           else{  return false;}}}

   public static void main(String[] args) {
      Scanner input = new Scanner(System.in);
      System.out.print("Enter text: ");
      String text = input.nextLine();
      if(PalCheck(text)){
         System.out.println(text+" is a palindrome");}
      else{   System.out.println(text+" is not a palindrome");}}}
```

Trial:

1) Enter text: dad
   dad is a palindrome

2) Enter text: catalog
   catalog is not a palindrome

10.Factorial_recursive

Code:

```
import java.util.Scanner;
public class Factorial_recursive {
   public static long fact(long n){     //uses recursion to calculate factorial
      if(n==0){return 1;}
      else{    return n*fact(n-1);}}
   public static void main(String[] args) {
      Scanner input = new Scanner(System.in);
      System.out.print("Enter number: ");
      long num = input.nextLong();
      System.out.println(fact(num));}}
```

Trial:

1) Enter number: 6

2) Enter number: 37 (PTO)

11.Dec2Base_recursive

Code:

```
import java.util.Scanner;
public class Dec2Base_recursive {
   public static void DecToBase(int n,int b){   //converts any decimal number to any base
      StringBuffer sb = new StringBuffer();
      char[] a = {'0','1','2','3','4','5','6','7','8','9','A','B','C','D','E','F',};  //elements correspond
      if(n<b){ sb.append(a[n]);}                                                       //to increasing powers
      else{ DecToBase((n/b),b);
          sb.append(a[(n%b)]);}        //uses StringBuffer as it is mutable
      System.out.print(sb);}

   public static void main(String[] args) {
      Scanner input = new Scanner(System.in);
      System.out.print("Enter number: "); int num = input.nextInt();
      System.out.print("Enter base: "); int base = input.nextInt();
      DecToBase(num,base);}}
```

Trial:

1) Enter number: 37
   Enter base: 16
   25

2) Enter number: 37
   Enter base: 2
   100101

3) Enter number: 37
   Enter base: 7
   52

12.PermutationCombination

Code:

```
import java.util.Scanner;
public class PermutationCombination {    //calculates permutations and combinations
   public static long factorial(long n){      //handy factorial tool
      if(n==0){ return 1;}
      else{ return n*factorial(n-1);}}

   public static void main(String[] args) {
      Scanner input = new Scanner(System.in);
      System.out.print("Enter 'p' for permutations or 'c' for combinations: ");
      char ch = input.next().charAt(0);
      System.out.print("Enter n value: ");
      long n = input.nextLong(); long n1 = n;
```

```
      System.out.print("Enter r value: "); long r = input.nextLong();
      long r1 = r; long nr = n-r; long fnr = factorial(nr);//n-r factorial
      long fn = factorial(n);   //n factorial
      long fr = factorial(r);   //r factorial
      long p = (fn)/(fnr); long c = p/fr; //preparation
      switch(ch) {  //gives user option btw p and c
         case 'p' : System.out.println(n1+"P"+r1+" = " + p);break;
         case 'c' : System.out.println(n1+"C"+r1+" = " + c);}}}
```

Trial:

```
1) Enter 'p' for permutations or 'c' for combinations: p
    Enter n value: 7
    Enter r value: 4
    7P4 = 840
2) Enter 'p' for permutations or 'c' for combinations: c
    Enter n value: 7
    Enter r value: 4
    7C4 = 35
```

---

13. Quadratic

Code:

```
import java.util.Scanner;
public class Quadratic {
   public static String solveQuad(double a,double b,double c) {
      double d2 = ((b*b)-(4*a*c)); //necessary calculations
      double d1 = Math.sqrt(d2);
      double d = d1/(2*a);
      double e = -b/(2*a); String x;
      if((e+d)==(e-d)){
         x = ("x is equal to "+(e+d));}
      else{  x = ("x is equal to "+(e+d)+" or "+(e-d));}
      return(x);}

   public static void main(String[] args) {
      Scanner input = new Scanner(System.in);
      System.out.println("For quadratic equation (ax^2 + bx + c)...");
      System.out.print("Enter 'a' value: ");          //gets user input
      double a = input.nextDouble();
      System.out.print("Enter 'b' value: ");
      double b = input.nextDouble();
```

```
      System.out.print("Enter 'c' value: ");
      double c = input.nextDouble();
      System.out.println(solveQuad(a,b,c));}}       //solves...
```

Trial:

1) For quadratic equation (ax^2 + bx + c)…
   Enter 'a' value: 1
   Enter 'b' value: -6
   Enter 'c' value: 9
   x is equal to 3.0

2) For quadratic equation (ax^2 + bx + c)…
   Enter 'a' value: 6
   Enter 'b' value: -13
   Enter 'c' value: -5
   x is equal to 2.5 or -0.3333333333333335

---

14. Quadrilateral

Code:

```
import java.util.*;
public class Quadrilateral {
   double perimeter,area; String type;
   public Quadrilateral(double len,double wid){          //new Class made
      perimeter = 2*(len+wid); area = len*wid;       //Calculation of area, type and
      if(len==wid){ type = "Square";}                //perimeter of a quadrilateral
      else type = "Rectangle";}

   public static void main(String[] args) {
      Scanner input = new Scanner(System.in);
      System.out.print("Enter length: ");
      double l = input.nextDouble();
      System.out.print("Enter width: ");
      double w = input.nextDouble();
      Quadrilateral quad = new Quadrilateral(l,w);
      System.out.println("The given quadrilateral is a "+quad.type+" of perimeter:
"+quad.perimeter+" units and area: "+quad.area+" unit^2.");}}
```

Trial:

1) Enter length: 3
   Enter width: 2
   The given quadrilateral is a Rectangle of perimeter: 10.0 units and area: 6.0 unit^2.
2) Enter length: 4
   Enter width: 4
   The given quadrilateral is a Square of perimeter: 16.0 units and area: 16.0 unit^2.

15.Write2Text

Code:

```
import java.io.*; import java.util.*;          //uses java.io library
public class Write2textFile {
   Scanner input = new Scanner(System.in);        //creates text file and stores a message
   String s = input.next();
   static String filename = ("NewText.txt");
   static BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
   public static void main(String[] args) {
      try{PrintWriter file = new PrintWriter(new BufferedWriter(new FileWriter(filename)));
         System.out.print("Enter text: ");
         String text = br.readLine();
         System.out.print("Your file is stored in Home/NetbeansProjects/Class11.");
         file.print(text); file.close();}
      catch(IOException e){
         System.out.println("");
         System.err.println(e);}}}
```

Trial:

```
Enter text: Delete this message
Your file is stored in Home/NetbeansProjects/Class11.
```

16.UserDefinedArray

Code:

```
import java.util.*;
public class UserDefinedArray {              //Creates array from user input & displays
   public static void main(String[] args) {
      Scanner input = new Scanner(System.in);
      System.out.print("Enter size of Int array:");
      int len = input.nextInt(); int[] array = new int[len];
      for(int h=0;h<len;h++){
         System.out.print("Enter Element "+(h+1)+":");
         int element = input.nextInt();
         array[h] = element;}
      for(int k=0;k<len;k++){
         System.out.print(array[k]);}}}
```

Trial:

```
Enter size of Int array:5
```

(PTO)

```
Enter Element 1:1
Enter Element 2:2
Enter Element 3:3
Enter Element 4:4
Enter Element 5:5
12345
```

---

17. **UserDefined2dArray**

Code:

```
import java.util.*;
public class UserDefined2dArray {          //Creates a 2D array from user input & displays
   public static void main(String[] args) {
      Scanner input = new Scanner(System.in);
      System.out.print("Enter number of arrays:"); int num = input.nextInt();
      System.out.print("Enter length of each array:"); int len = input.nextInt();
      int[][] array = new int[num][len];
      for(int i=0;i<num;i++){
         System.out.println("For array "+(i+1)+":");
         for(int j=0;j<len;j++){
            System.out.print("Enter element "+(j+1)+" of array "+(i+1)+":");
            array[i][j] = input.nextInt();}}
      for(int i=0;i<num;i++){
         for(int j=0;j<len;j++){
            System.out.print(array[i][j]+" ");}
         System.out.println();}}}
```

Trial:

```
Enter number of arrays:3
Enter length of each array:2
For array 1:
Enter element 1 of array 1:1
Enter element 2 of array 1:2
For array 2:
Enter element 1 of array 2:3
Enter element 2 of array 2:4
For array 3:
Enter element 1 of array 3:5
Enter element 2 of array 3:6
```

(PTO)

```
1 2
3 4
5 6
```

---

18. **BubbleSort**

Code:

```
import java.util.Scanner;
public class Bubble_Sort {
   public static void BubbleSort(int a[], int len){          //uses Bubble Sort method to sort
      int i,j,k,t;                                           //array elements in increasing order
      for(i=0;i<len;i++){
         for(j=0;j<(len-(i+1));j++){
            if(a[j]>a[j+1]){
               t=a[j]; a[j]=a[j+1]; a[j+1]=t;}}}
      for(k=0;k<len;k++){
         System.out.print(a[k]);}}

   public static void main(String[] args) {
      Scanner input = new Scanner(System.in);
      System.out.print("Enter size of Int array:");
      int len = input.nextInt(); int[] array = new int[len];
      for(int h=0;h<len;h++){
         System.out.print("Enter Element "+(h+1)+":");
         int element = input.nextInt(); array[h] = element;}
      BubbleSort(array, len);}}
```

Trial:

```
1) Enter size of Int array:6
   Enter Element 1:4
   Enter Element 2:2
   Enter Element 3:7
   Enter Element 4:5
   Enter Element 5:9
   Enter Element 6:1
   124579
```

```
2) Enter size of Int array:7
   Enter Element 1:4
   Enter Element 2:6
   Enter Element 3:1
   Enter Element 4:9
   Enter Element 5:5
   Enter Element 6:3
   Enter Element 7:1
   1134569
```

19.**SelectionSort**

Code:

```
import java.util.Scanner;
public class Selection_Sort {
   public static void SelectionSort(int a[], int len){        //uses Selection Sort method to sort
      int i,j,k,smallt,t,position;                            //array elements in increasing order
      for(i=0;i<len;i++){
         smallt = a[i]; position=i;
         for(j=i+1;j<len;j++){
            if(a[j]<smallt){
               smallt = a[j]; position = j;}}
         t=a[i]; a[i]=a[position]; a[position]=t;}
      for(k=0;k<len;k++){
         System.out.print(a[k]);}}

   public static void main(String[] args) {
      Scanner input = new Scanner(System.in);
      System.out.print("Enter size of Int array:");
      int len = input.nextInt(); int[] array = new int[len];
      for(int h=0;h<len;h++){
         System.out.print("Enter Element "+(h+1)+": ");
         int element = input.nextInt(); array[h] = element;}
      SelectionSort(array, len);}}
```

Trial:

1) Enter size of Int array:6
   Enter Element 1: 4
   Enter Element 2: 2
   Enter Element 3: 3
   Enter Element 4: 1
   112349

2) Enter size of Int array:4
   Enter Element 1: 2
   Enter Element 2: 9
   Enter Element 3: 8
   Enter Element 4: 1
   1289

20.**Matrix_Addition**

Code:

```
import java.util.*;
public class Matrix_Addition {                  //Creates 2 matrices from user input
   public static void display(int[][] m){       //and prints their sum
      for(int i=0;i<m.length;i++){
         for(int j=0;j<m[i].length;j++){
```

```
            System.out.print(m[i][j]+" ");}     //Displays the final/sum matrix
        System.out.println();}}

    public static void add(int[][] m, int[][] m1,int[][] m2){
        System.out.println(" =");
        for(int i=0;i<m.length;i++){
            for(int j=0;j<m[i].length;j++){         //Adds the 2 matrices
                m[i][j] = m1[i][j] + m2[i][j];
                System.out.print(m[i][j]+ " ");}
            System.out.println();}}

    public static void getMatrix(int[][] m,int n){              //Creates matrices from user input
        Scanner input = new Scanner(System.in);
        System.out.println("For matrix "+n+": ");
        for(int i=0;i<m.length;i++){
            for(int j=0;j<m[i].length;j++){
                System.out.print("Enter element "+(j+1)+" of row "+(i+1)+":");
                m[i][j] = input.nextInt();}}}

    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        System.out.print("Enter number of rows per matrix: ");
        int rows = input.nextInt();
        System.out.print("Enter number of columns per matrix: ");
        int columns = input.nextInt();
        int[][] m1 = new int[rows][columns]; getMatrix(m1,1);
        int[][] m2 = new int[rows][columns]; getMatrix(m2,2);
        int[][] m = new int[rows][columns]; System.out.println();
        display(m1);System.out.println(" +");display(m2);add(m,m1,m2);}}
```

Trial:

```
Enter number of rows per matrix: 2
Enter number of columns per matrix: 3
For matrix 1:
Enter element 1 of row 1:1
Enter element 2 of row 1:2
Enter element 3 of row 1:3
Enter element 1 of row 2:4
Enter element 2 of row 2:5
Enter element 3 of row 2:6
For matrix 2:
```

```
Enter element 1 of row 1:8
Enter element 2 of row 1:7
Enter element 3 of row 1:6
Enter element 1 of row 2:3
Enter element 2 of row 2:2
Enter element 3 of row 2:3

1 2 3
4 5 6
 +
8 7 6
3 2 3
 =
9 9 9
7 7 9
```

21.**Matrix_Transposal**

Code:

```
import java.util.*;
public class Matrix_Transposal {

   public static void transpose(int[][]m1,int[][]m2){      //converts rows to columns and vice versa
      for(int i=0;i<m1.length;i++){
         for(int j=0;j<m1[i].length;j++){
            m2[i][j] = m1[j][i];}}
      System.out.println("Converted to");}

   public static void display(int[][] m){              //Displays the final/sum matrix
      System.out.println();
      for(int i=0;i<m.length;i++){
         for(int j=0;j<m[i].length;j++){
            System.out.print(m[i][j]+" ");}
         System.out.println();}System.out.println();}

   public static void getMatrix(int[][] m){            //Creates matrices from user input
      Scanner input = new Scanner(System.in);
      for(int i=0;i<m.length;i++){
         for(int j=0;j<m[i].length;j++){
            System.out.print("Enter element "+(j+1)+" of row "+(i+1)+":");
            m[i][j] = input.nextInt();}}}
```

```
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        System.out.print("Enter number of rows in square matrix: ");
        int n = input.nextInt();
        int[][] m = new int[n][n];int[][] mat = new int[n][n];
        getMatrix(m);display(m);transpose(m,mat);display(mat);}}
```

Trial:

```
Enter number of rows in square matrix: 3
Enter element 1 of row 1:1
Enter element 2 of row 1:2
Enter element 3 of row 1:3
Enter element 1 of row 2:4
Enter element 2 of row 2:5
Enter element 3 of row 2:6
Enter element 1 of row 3:7
Enter element 2 of row 3:8
Enter element 3 of row 3:9

1 2 3
4 5 6
7 8 9

Converted to

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