**TECHNICAL PROGRAMMING 1**

**2016 SUPPLEMENTARY EXAM MEMORANDUM**

**QUESTION ONE [15 MARKS]**

**TRUE/FALSE**

1.1. ANS: T REF: 44

1.2. ANS: F REF: 44

1.3. ANS: F REF: 50

1.4. ANS: T REF: 64, 65

1.5. ANS: F REF: 174, 175

1.6. ANS: F REF: 180

1.7. ANS: T REF: 226

1.8. ANS: T REF: 229

1.9. ANS: T REF: 231

1.10.ANS: F REF: 260

1.11.ANS: T REF: 273

1.12.ANS: F REF: 367

1.13.ANS: T REF: 537

1.14.ANS: T REF: 54

1.15.ANS: F REF: 552

**QUESTION TWO** **[20 MARKS]**

**MULTIPLE CHOICE**

2.1. ANS: C REF: 46

2.2. ANS: C REF: 50

2.3. ANS: D REF: 67

2.4. ANS: C REF: 193

2.5. ANS: C REF: 190, 194, 195

2.6. ANS: D REF: 200-202

2.7. ANS: C REF: 225, 226

2.8. ANS: C REF: 231, 232

2.9. ANS: A REF: 253

2.10. ANS: C REF: 258-260

2.11. ANS: A REF: 259, 260

2.12. ANS: C REF: 258-260

2.13. ANS: D REF: 268, 269

2.14. ANS: D REF: 268, 269

2.15. ANS: B REF: 268, 269

2.16. ANS: C REF: 552

2.17. ANS: C REF: 570, 571

2.18. ANS: D REF: 572

2.19. ANS: C REF: 572

2.20. ANS: D REF: 572

**QUESTION THREE [7 MARKS]**

import javax.swing.\*;  
class Palindrome   
{  
 public static void main(String[]args)   
 {  
 String word = "";  
 String reversedWord = "";  
   
 word = JOptionPane.showInputDialog(null,"Please enter a word:");  
 int len = word.length();  
   
  
 for(int i = len-1; i >= 0; i--)  
 { reversedWord = reversedWord + word.charAt(i);  
 }   
   
 if (word.equals(reversedWord))  
 JOptionPane.showMessageDialog(null," The word " + word + " is a palindrome.");  
  
 else   
 JOptionPane.showMessageDialog(null," The word " + word + " is NOT a palindrome, \n as spelt backwards it reads " + reversedWord);  
 }  
}

**QUESTION FOUR [30 MARKS]**

import java.util.\*;  
public class Pageant  
{  
 public static void main(String []args)  
 { String [] contestants = new String[12];  
 int [][] scores = new int[12][3];  
 double [] average = new double[12];

inputContestants(contestants);  
 inputScores(scores, contestants);  
 calculateAverage(scores, average);  
 findHighestAverage(average, contestants);  
 displayResults(contestants, scores, average);  
 searchResults(contestants, average);  
 }  
   
 public static void inputContestants(String []contestants)  
 { Scanner kb = new Scanner(System.in);   
 int index;

for(index=0; index<12; index++)  
 { System.out.println("Enter the name for contestant :" + (index+1));   
 contestants[index] = kb.nextLine();  
 }   
 }  
  
 public static void inputScores(int [][] scores, String[]contestants)  
 { Scanner kb = new Scanner(System.in);   
 int c, j;

for(c=0; c<12; c++)  
 { System.out.println("Contestant is:" + contestants[c]);

for(j=0; j<3; j++)  
 { System.out.println("Enter score from judge " + (j+1));  
 scores[c][j] = kb.nextInt();  
 }  
 }   
 }  
   
 public static void calculateAverage(int [][] scores, double [] average)  
 { int c, j, sum;

for(c=0; c<12; c++)  
 { sum = 0;   
   
 for(j=0; j<3; j++)  
 sum = sum + scores[c][j];  
   
 average[c] = sum / 3;  
 }  
 }

public static void findHighestAverage(double[]average, String[] contestants)  
 { int c, best = 0;  
 double highest=0;  
   
 for(c=0; c<12; c++)  
 {   
 if (average[c] > highest)  
 { highest = average[c];  
 best = c;  
 }  
 }

System.out.println("The winner of Miss SA is " + contestants[best] + " \nwith the highest average of " + highest);  
 }

public static void displayResults(String [] contestants, int [][]scores, double [] average)  
 { int c, j;  
 System.out.println("Contestant \t Judge 1\tJudge 2\t\tJudge 3\t Average");

for(c=0; c<12; c++)  
 { System.out.print(contestants[c] + "\t\t");

for(j=0; j<3; j++)  
 { System.out.print(scores[c][j] + "\t\t");}

System.out.print(average[c]);   
 System.out.println();   
 }  
 }  
  
 public static void searchResults(String [] contestants, double[]average)  
 { int c = 0;   
 boolean found = false;   
 Scanner kb = new Scanner(System.in);

System.out.println("Enter name of contestant:");  
 String name = kb.nextLine();

while (!found && c < contestants.length)  
 { if (name.equalsIgnoreCase(contestants[c]))  
 found = true;  
 else  
 c++;  
 }

if (found==true)  
 System.out.println("Average score is: " + average[c]);  
 else  
 System.out.println("Contestant was not found!");   
 }

}

**QUESTION FIVE [28 MARKS]**

**BOOK CLASS (22)**

public class Book  
{

private String title;  
 private String author;  
 private double price;  
 private int copies;  
   
 public Book()  
 {

title = "":

author = "";

price = 0;

copies = 0;  
 }  
   
 public Book(String t, String a, double p, int c)  
 {

title = t;  
 author = a;  
 price = p;  
 copies = c;  
 }  
   
 public void setAuthor(String a)  
 {

author = a;   
 }  
   
 public void setTitle(String t)  
 {

title = t;   
 }  
  
 public void setPrice(double p)  
 {

price = p;   
 }  
  
 public void setCopies(int c)  
 {

copies = c;   
 }

public String getTitle()  
{

return title;  
 }  
  
 public String getAuthor()  
 {

return author;  
 }  
  
 public double getPrice()  
 {

return price;  
 }

public int getCopies()  
 {

return copies;  
}

public void updateCopies(int c)  
{

copies = copies + c;  
}  
   
public void increasePrice()  
{

price = price + (price \* 0.10);  
}  
   
public String toString()  
{

String outString = "";  
outString = "Title: " + title + "\nAuthor: " + author + "\nPrice of book: " + price + "\nNo of copies available: " + copies;  
return outString;

}

}

**TEST DRIVER (6)**import java.util.\*;  
  
public class testBook  
{ public static void main(String [] args)  
 { Scanner kb = new Scanner(System.in);  
   
 String title, author;  
 double price;  
 int copies;  
   
 Book fiction = new Book();  
   
 System.out.println("Enter book title:");  
 title = kb.nextLine();  
   
 System.out.println("Enter book author:");  
 author = kb.nextLine();  
  
 System.out.println("Enter price of book:");  
 price = kb.nextDouble();  
   
 System.out.println("Enter number of copies available:");  
 copies = kb.nextInt();  
   
 Book nonfiction = new Book(title, author, price, copies);  
   
 System.out.println(fiction.toString());   
 System.out.println(nonfiction.toString());  
 }  
  
}