|  |
| --- |
| **Sheet 1,pl2**[إياد امين احمد احمد الفار] |
| 202000172 |

**1.5 Write a program that displays the result of 9.5 \* 4.5 - 2.5 \* 3 / 45.5 - 3.5 :**

Sol:

class main{

        //9.5 \* 4.5 - 2.5 \* 3 / 45.5 - 3.5 .

    public static void main(String args[]) {

    System.out.println(((9.5\*4.5)-(2.5\*3)) / (45.5 - 3.5));

    }

}

**1.9 (*Area and perimeter of a rectangle*) Write a program that displays the area and**

**perimeter of a rectangle with a width of 4.5 and a height of 7.9 using the following formula:**

*area* = *width* \* *height*

Sol:

class main{

    public static void main(String args[]) {

    double width,height,area;

    height=7.9;

    width=4.5;

    area = width\*height;

    System.out.println(area);

    }

}

**\*1.11**

(*Population projection*) The U.S. Census Bureau projects population based on

the following assumptions:

■ One birth every 7 seconds

■ One death every 13 seconds

■ One new immigrant every 45 seconds

Write a program to display the population for each of the next five years. Assume

that the current population is 312,032,486, and one year has 365 days. *Hint*: In Java,

if two integers perform division, the result is an integer. The fractional part is truncated. For example, **5** / **4** is **1** (not **1.25**) and **10** / **4** is **2** (not **2.5**). To get an accurate result with the fractional part, one of the values involved in the division must be

a number with a decimal point. For example, **5.0** / **4** is **1.25** and **10** / **4.0** is **2.5**.

Sol:

class main{

    public static void main(String args[]) {

    int pop = 312032486,yearInSec= 365\*24\*60\*60;

    // 1 year in seconds is 365\*24\*60\*60

    int birth= yearInSec/7;

    int death= yearInSec/13;

    int immigrant= yearInSec/45;

    int newpop= (birth+immigrant) - death;

    for(int i=0;i<5;i++){

        pop+= newpop;

        System.out.println("#"+ (i+1) + " year/s population is: " +pop);}}}

**Chapter 2:**

**2.2**

(*Compute the volume of a cylinder*) Write a program that reads in the radius

and length of a cylinder and computes the area and volume using the following

formulas:

area = radius \* radius \* π

volume = area \* length

Sol:

class main{

    public static void main(String args[]) {

    Scanner Cylinder = new Scanner(System.in);

    double radius,length,area,volume;

    System.out.println("Enter the radius and the length of the cylinder: ");

    radius=Cylinder.nextDouble();

    area=radius\*radius\*3.14;

    length=Cylinder.nextDouble();

    volume=area\*length;

    System.out.println("The area is : " + area);

System.out.println("The volume is : " + volume); }}

**\*2.5**

(*Financial application: calculate tips*) Write a program that reads the subtotal

and the gratuity rate, then computes the gratuity and total. For example, if the

user enters **10** for subtotal and **15%** for gratuity rate, the program displays **$1.5**

as gratuity and **$11.5** as total.

Sol:

import java.util.Scanner;

class main{

    public static void main(String args[]) {

        double subtotal,gratuityrate,total,gratuity;

        Scanner calc = new Scanner(System.in);

        System.out.println("Enter the gratuity rate and subtotal : ");

        subtotal = calc.nextDouble();

        gratuityrate = calc.nextDouble();

        gratuity=gratuityrate\*subtotal/100;

        total=subtotal+gratuity;

        System.out.println("The gratuity is: " + gratuity +" and total is $ " + total ); }}

**\*\*2.6**

(*Sum the digits in an integer*) Write a program that reads an integer between **0**

and **1000** and adds all the digits in the integer. For example, if an integer is **932**,

the sum of all its digits is **14**.

*Hint*: Use the **%** operator to extract digits, and use the **/** operator to remove the

extracted digit. For instance, **932 % 10 = 2** and **932 / 10 = 93**.

Sol:

import java.util.Scanner;

class main{

    public static void main(String args[]) {

        int sum,temp;

        Scanner num = new Scanner(System.in);

        System.out.println("Enter a number between 0 and 1000 : ");

        temp = num.nextInt();

        sum = temp % 10;

        temp /= 10;

        sum += temp % 10;

        temp /= 10;

        sum += temp % 10;

        System.out.println("The sum of the digits is: " + sum );

     }}

**\*\*3.4**

(*Random month*) Write a program that randomly generates an integer between 1

and 12 and displays the English month names January, February, . . . , December

for the numbers 1, 2, . . . , 12, accordingly.

Sol:

import java.util.Random;

public class aaa {public static void main(String args[]) {

        Random month = new Random();

        int num = 1+month.nextInt(12);

        System.out.println(num);

        switch(num){

            case(1):

                System.out.println("January");

                break;

            case(2):

                System.out.println("February");

                break;

            case(3):

                System.out.println("March");

                break;

            case(4):

                System.out.println("April");

                break;

            case(5):

                System.out.println("May");

                break;

            case(6):

                System.out.println("June");

                break;

            case(7):

                System.out.println("July");

                break;

            case(8):

                System.out.println("August");

                break;

            case(9):

                System.out.println("September");

                break;

            case(10):

                System.out.println("October");

                break;

            case(11):

                System.out.println("November");

                break;

            case(12):

                System.out.println("December");

                break; } } }

**\*3.5**

(*Find future dates*) Write a program that prompts the user to enter an integer for

today’s day of the week (Sunday is 0, Monday is 1, . . . , and Saturday is 6). Also

prompt the user to enter the number of days after today for a future day and dis

play the future day of the week.

Sol:

import java.util.Scanner;

public class aaa {

    public static void main(String args[]) {

        Scanner day = new Scanner(System.in);

        int today,dayselapsed,futureday;

        String dayinweek="";

        System.out.println("Enter today's day, ");

        System.out.println("For ex. Sunday is 0,monday is 1 .. saturday is 6 :");

        today=day.nextInt();

        System.out.println("Enter the number of days elapsed since today :");

        dayselapsed=day.nextInt();

        futureday=(today+dayselapsed) % 7;

        switch(today){

            case(0):

            dayinweek="Sunday";

            break;

            case(1):

            dayinweek="Monday";

            break;

            case(2):

            dayinweek="Teusday";

            break;

            case(3):

            dayinweek="Wednesday";

            break;

            case(4):

            dayinweek="Thursday";

            break;

            case(5):

            dayinweek="Friday";

            break;

            case(6):

            dayinweek="Saturday";

            break;

        }

        if (futureday== 0){

            System.out.println("Todays is " + dayinweek + " and the future day is Sunday");

        }else if(futureday == 1){

            System.out.println("Todays is " + dayinweek + " and the future day is Monday");

        }else if(futureday == 2){

            System.out.println("Todays is " + dayinweek + " and the future day is Tuesday");

        }else if(futureday == 3){

            System.out.println("Todays is " + dayinweek + " and the future day is Wednesday");

        }else if(futureday == 4){

            System.out.println("Todays is " + dayinweek + " and the future day is Thursday");

        }else if(futureday == 5){

            System.out.println("Todays is " + dayinweek + " and the future day is Friday");

        }else if(futureday == 6){

            System.out.println("Todays is " + dayinweek + " and the future day is Saturday");

        }

        //sunday = 0 , or sunday = 0+7d if d = 0 then sunday = 0 if d =1 then sunday =7

        // we can write it as if 31%7 = 3 , wednesday, 32%7=4 thursday

     } }

**\*\*3.9**

(*Business: check ISBN-10*) An ISBN-10 (International Standard Book Number)

consists of 10 digits: *d*1*d*2*d*3*d*4*d*5*d*6*d*7*d*8*d*9*d*10. The last digit, *d*10, is a checksum,

which is calculated from the other 9 digits using the following formula:

(*d*1 \* 1 + *d*2 \* 2 + *d*3 \* 3 + *d*4 \* 4 + *d*5 \* 5 +

*d*6 \* 6 + *d*7 \* 7 + *d*8 \* 8 + *d*9 \* 9)%11

If the checksum is **10**, the last digit is denoted as X according to the ISBN-10

convention. Write a program that prompts the user to enter the first 9 digits and

displays the 10-digit ISBN (including leading zeros). Your program should read

the input as an integer.

Sol:

import java.util.Scanner;

public class aaa {

    public static void main(String args[]) {

        Scanner digits = new Scanner(System.in);

        int d1,d2,d3,d4,d5,d6,d7,d8,d9,d10,alldigits,temp;

        System.out.print("Enter the first 9 digits of an ISBN as integer: ");

        alldigits=digits.nextInt();

        temp=alldigits;

        d9=temp%10; temp/=10; d8=temp%10; temp/=10;

        d7=temp%10; temp/=10; d6=temp%10; temp/=10;

d5=temp%10; temp/=10; d4=temp%10; temp/=10;

        d3=temp%10; temp/=10; d2=temp%10; temp/=10;

        d1=temp%10;

        d10=(d1\*1 + d2\*2 + d3\*3 + d4\*4 + d5\*5 + d6\*6 + d7\*7 + d8\*8 + d9\*9) % 11;

        if(d10 != 10) {System.out.printf("%09d%d",alldigits,d10); }

        else { System.out.printf("%09dX",alldigits); } } }

**\*4.2** (*Geometry: great circle distance)* The great circle distance is the distance be

tween two points on the surface of a sphere. Let (*x*1, *y*1) and (*x*2, *y*2) be the

geographical latitude and longitude of two points. The great circle distance be

tween the two points can be computed using the following formula:

*d* = *radius* \* arccos(sin (*x*1) \* sin(*x*2) + cos(*x*1) \* cos(*x*2) \* cos(*y*1 - *y*2))

Write a program that prompts the user to enter the latitude and longitude of two

points on the earth in degrees and displays its great circle distance. The average

radius of the earth is 6,371.01 km. Note you need to convert the degrees into

radians using the **Math.toRadians** method since the Java trigonometric meth

ods use radians. The latitude and longitude degrees in the formula are for north

and west. Use negative to indicate south and east degrees.

Sol: ??

import java.util.Scanner;

public class aaa {

    public static void main(String args[]) {

        Scanner latlong = new Scanner(System.in);

        double p1lat,p1long,p2lat,p2long,radius,d,simplfy;

        radius= 6371.01;

        System.out.print("Enter point 1 (latitude and longitude) in degrees: ");

        p1lat = latlong.nextDouble();

        p1lat = Math.toRadians(p1lat);

        p1long = latlong.nextDouble();

        p1long = Math.toRadians(p1long);

        System.out.print("Enter point 2 (latitude and longitude) in degrees: ");

        p2lat = latlong.nextDouble();

        p2lat = Math.toRadians(p2lat);

        p2long = latlong.nextDouble();

        p2long = Math.toRadians(p2long);

        simplfy= Math.acos( (Math.sin(p1lat)\*Math.sin(p1long)) + Math.cos(p1lat)\*Math.cos(p1long)\*Math.cos(p2lat-p2long));

        d=radius\* simplfy;

        System.out.printf("\nThe distance between the two points is %f km\n",d); } }

**\*4.3**

(*Geography: estimate areas*) Use the GPS locations for Atlanta, Georgia;

Orlando, Florida; Savannah, Georgia; and Charlotte, North Carolina in the figure in

Section 4.1 to compute the estimated area enclosed by these four cities.

(Hint: Use the formula in Programming Exercise 4.2 to compute the distance

between two cities. Divide the polygon into two triangles and use the formula in

Programming Exercise 2.19 to compute the area of a triangle.)

Sol:???

**\*4.5**

(*Geometry: area of a regular polygon*) A regular polygon is an *n*-sided polygon

in which all sides are of the same length and all angles have the same degree (i.e.,

the polygon is both equilateral and equiangular). The formula for computing the

area of a regular polygon is:

*Area* = (*n* \* *s2* ) / (4 \* tan(pi/*n* )) Here, *s* is the length of a side. Write a program that prompts the user to enter the number of sides and their length of a regular polygon and displays its area.

Sol:

import java.util.Scanner;

public class aaa {

    public static void main(String args[]) {

        Scanner sides = new Scanner(System.in);

        double s,n,Area;

        System.out.print("Enter the number of sides: ");

        n = sides.nextDouble();

        System.out.print("Enter the side length: ");

        s = sides.nextDouble();

        Area =(n \* Math.pow(s, 2)) / (4 \* Math.tan(Math.PI/n));

        System.out.printf("\nThe area of the polygon is: %f",Area);

         } }