**Program : 1**

object MyClass {

def q1() {

println("Name - Aadarsh Agarwal");

println("Sec- DS");

println("Roll No. - 2014501");

println("University Roll No. - 19021645");

println("enter temperature");

var fahrenheit = scala.io.StdIn.readFloat();

var celsius = ((fahrenheit - 32) \* 5) / 9;

print(celsius);

};

def q2() {

println("enter inches");

var inches = scala.io.StdIn.readFloat();

var meters = inches \* 0.0254;

print(meters);

}

def q3() {

println("enter no. of years");

var years = scala.io.StdIn.readInt();

print(years \* 365);

}

def main(args: Array[String]) {

q1();

q2();

q3();

}

}

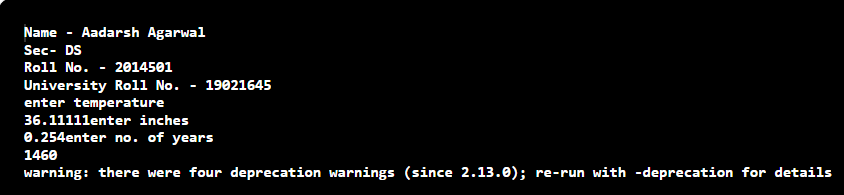
**Input :**

97

10

4

**Output :**



**Program : 2**.

object MyClass {

def main(args: Array[String]) {

println("Name - Aadarsh Agarwal");

println("Sec- DS");

println("Roll No. - 2014501");

println("Enrollment No. - 19021645");

var add = (x: Int, y: Int) => x + y;

var sub = (x: Int, y: Int) => x - y;

var multi = (x: Int, y: Int) => x \* y;

var divide = (x: Int, y: Int) => x / y;

println("1.add");

println("2.sub");

println("3.multiply");

println("4.divide");

println("Enter choice");

var choice = scala.io.StdIn.readInt();

println("Enter X and Y");

var a = scala.io.StdIn.readInt();

var b = scala.io.StdIn.readInt();

if (choice == 1) {

var result = add(a, b);

println(result);

} else if (choice == 2) {

var result = sub(a, b);

println(result);

} else if (choice == 3) {

var result = multi(a, b);

println(result);

} else if (choice == 4) {

var result = divide(a, b);

println(result);

}

}

}

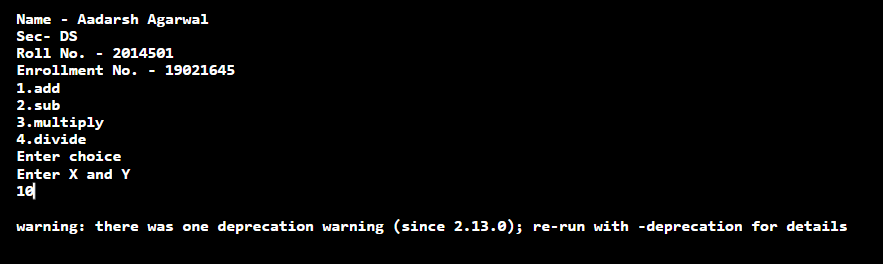
**Input :**

1

5

5

**Output :**



**Program : 3**

trait Car {

def horn()

def

break ()

def go()

}

trait Truck {

def size()

}

class BigCar extends Car {

def horn() {

println("horned")

}

def

break () {

println("breaked");

}

def go() {

println("going");

}

}

class CyberTruck extends Car with Truck {

def horn() {

println("peeeeep")

}

def

break () {

println("stopped");

}

def go() {

println("brrrr");

}

def size() {

println("big size");

}

}

object MyClass {

def main(args: Array[String]) {

println("Name - Aadarsh Agarwal");

println("Sec- DS");

println("Roll No. - 2014501");

println("Enrollment No. - 19021645");

var x = new CyberTruck();

var y = new BigCar();

x.horn();

x.break();

x.go();

x.size();

y.horn();

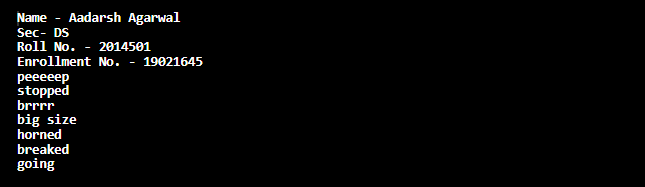
y.break();

y.go();

}

}

**Output :**



**Program : 4**

import scala.annotation.tailrec

object q4 {

@tailrec def fact(acc: Int, n: Int): Int = {

if (n <= 1)

acc

else

fact(n \* acc, n - 1)

}

@tailrec def gcd(x: Int, y: Int): Int = {

if (y == 0)

x

else

gcd(y, x % y)

}

def main(args: Array[String]) = {

println("Name - Aadarsh Agarwal");

println("Sec- DS");

println("Roll No. - 2014501");

println("Enrollment No. - 19021645");

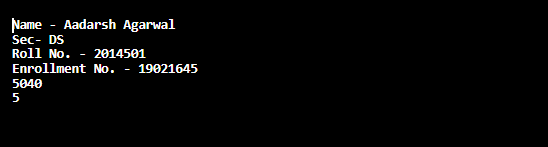
println(fact(1, 7))

println(gcd(5, 10))

}

}

**Output :**



**Program : 5**

object MyClass {

def main(args: Array[String]): Unit = {

println("Name - Aadarsh Agarwal");

println("Sec- DS");

println("Roll No. - 2014501");

println("Enrollment No. - 19021645");

println("Creating List")

val lis = List(1, 2, 3, 4, 5, 6, 7, 8, 9, 1, 2, 3, 4, 5, 6)

println("List is")

println(lis)

println("Distinct Element list")

val distiner = (lis: List[Int]) => lis.distinct

val ans = distiner(lis)

println(ans)

var n = 9

println("First Occurance of n")

println(ans.indexOf(n))

println("Reversing a list")

val len = (lis: List[Int]) => {

println(lis.length)

println(lis.reverse)

}

len(ans)

println("maximum and minimum")

var min = (l: List[Int]) => l.min

var max = (l: List[Int]) => l.max

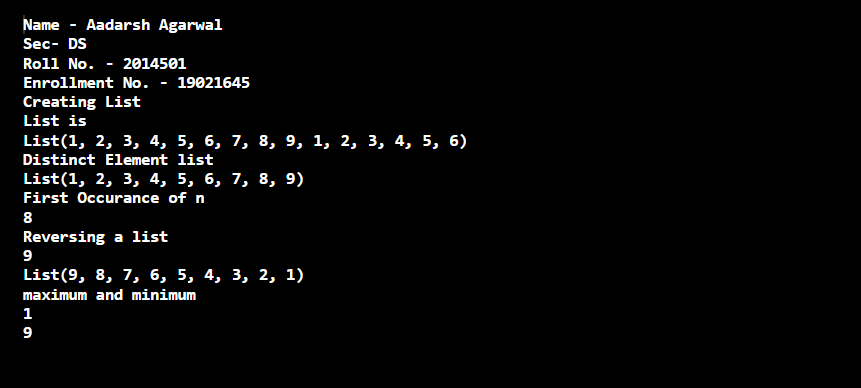
println(min(ans))

println(max(ans))

}

}

**Output :**



**Program : 6**

object MyClass {

def main(args: Array[String]): Unit = {

println("Name - Aadarsh Agarwal");

println("Sec- DS");

println("Roll No. - 2014501");

println("Enrollment No. - 19021645");

println("Creating List")

val lis = List(1, 2, 3, 4, 5, 6, 7, 8, 9, 1, 2, 3, 4, 5, 6)

println("List is")

println(lis)

println("Distinct Element list")

val distincter = (lis: List[Int]) => lis.distinct

val ans = distincter(lis)

println(ans)

var n = 9

println("First Occurance of n")

println(ans.indexOf(n))

println("Reversing a list")

val len = (lis: List[Int]) => {

println(lis.length)

println(lis.reverse)

}

len(ans)

println("maximum and minimum")

var min = (l: List[Int]) => l.min

var max = (l: List[Int]) => l.max

println(min(ans))

println(max(ans))

println(ans.zipWithIndex.map {

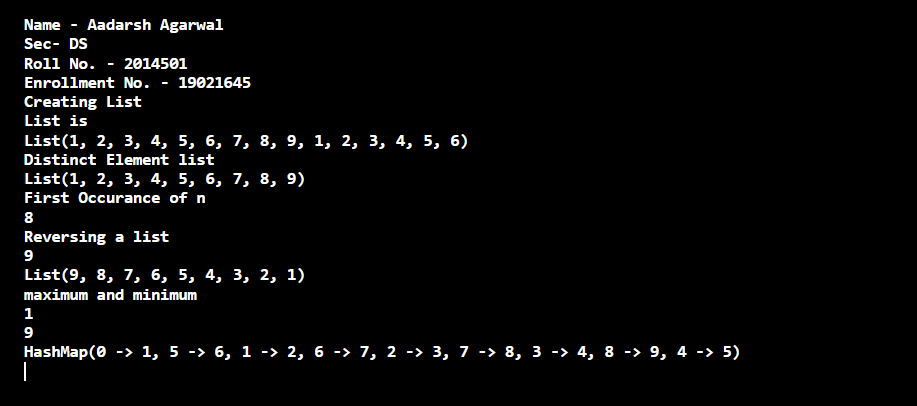
case (v, i) => (i, v)

}.toMap);

}

}

**Output :**



**Program : 7**

object MyClass {

def main(args: Array[String]) = {

println("Name - Aadarsh Agarwal");

println("Sec- DS");

println("Roll No. - 2014501");

println("Enrollment No. - 19021645");

var a = List(99.5, 100.0, 50.0, 55.0, 70.0, 100, -1.0);

var b = List(10.0, 20.0, 30.0, 40.0, 50.0);

var c = a::: b;

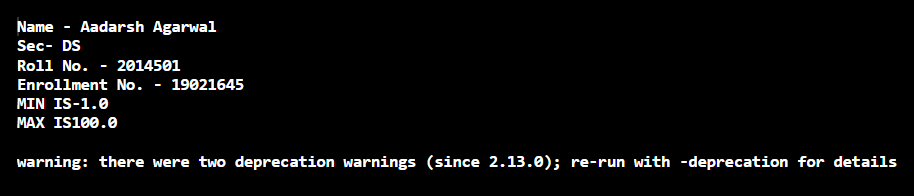
println("MIN IS" + c.min);

println("MAX IS" + c.max);

}

}

**Output :**



**Program : 8**

object q8 {

def main(args: Array[String]) {

println("Name - Aadarsh Agarwal");

println("Sec- DS");

println("Roll No. - 2014501");

println("Enrollment No. - 19021645");

var age = scala.io.StdIn.readInt();

var checkeligible = (age: Int) => age >= 18

if (checkeligible(age)) {

println("Eligible to vote");

} else {

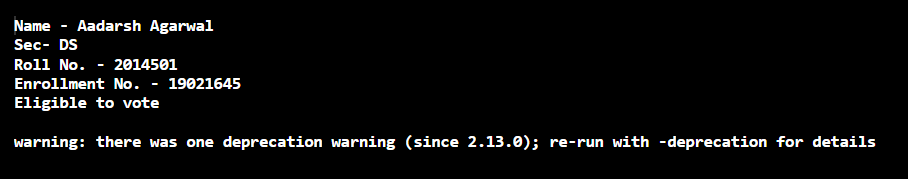
println("Not eligible to vote");

}

}

}

**Output :**



**Program : 9**

import scala.collection.immutable.ListMap

object MyClass {

def main(args: Array[String]): Unit = {

println("Name - Aadarsh Agarwal");

println("Sec- DS");

println("Roll No. - 2014501");

println("Enrollment No. - 19021645");

val ma = Map("Bill" -> 56,

"Jonny" -> 87,

"Tommy" -> 11,

"Cheena"-> 14

);

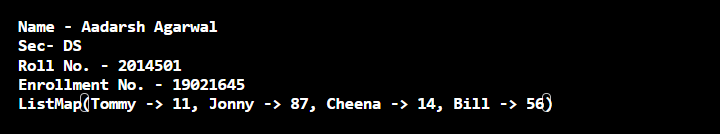
val res = ListMap(ma.toSeq.sortWith(\_.\_1 > \_.\_1):\_\*);

print(res);

}

}

**Output :**



**Program : 10**

object MyClass {

def main(args: Array[String]) {

println("Name - Aadarsh Agarwal");

println("Sec- DS");

println("Roll No. - 2014501");

println("Enrollment No. - 19021645");

val sen = scala.io.StdIn.readLine()

var n = sen.length()

if (n < 4) {

println("Your sentence in upper case: " + sen.toUpperCase())

} else {

println("Your sentence in upper case: " + sen.takeRight(4).toUpperCase())

}

}

}

**Output :**

