**Test1**

1. Create C# Console Application that shows multiplication table from 1 to 16.

using System;

class program

{

static void Main(string[] args)

{

for (int i = 1; i <= 16; i++)

{

for (int j = 1; j <= 12; j++)

{

Console.WriteLine(i + "x" + j + "=" + i \* j);

}

Console.WriteLine("----------------------");

}

}

}

2. Write a C# program that calculate the following equation with user input data.

F = 9/5 C + 32 (where F is the temperature in degrees Fahrenheit and C is the temperature in degrees Celsius.)

using System;

class program

{

static void Main(string[] args)

{

Console.WriteLine("Enter Degree in Celsius to calculate F = 9/5 C + 32 :");

double C = Convert.ToDouble(Console.ReadLine());

Console.WriteLine("Degree in Fahrenheit=" + (((9 / 5.0) \* C) + 32).ToString());

}

}

3. Write a C# Program that accepts student’s name and phone number by separating comma until user input request is yes. Then sort their names in **descending** order.

E.g, Aye Aye, 09250504556

using System;

using System.Collections;

class program

{

static void Main(string[] args)

{

string user = "";

var userlist = new ArrayList();

bool next = true;

while (next)

{

Console.WriteLine("Write your name and ph no by separating comma :");

user = Console.ReadLine();

userlist.Add(user);

Console.WriteLine("Next one? enter key for yes, no for other keys");

next = Console.ReadKey(true).Key == ConsoleKey.Enter ? true : false;

}

userlist.Sort(); //userlist.Reverse();

foreach ( var u in userlist)

Console.WriteLine(u);

}

}

4. Write C# Lucky draw program that generate first, second, and third prize for the users in question number 3. Show the prizes and user information.

using System;

using System.Collections;

using System.Collections.Generic;

class program

{

static void Main(string[] args)

{

string user = "";

var userlist = new ArrayList();

bool next = true;

while (next)

{

Console.WriteLine("Write your name and ph no by separating comma :");

user = Console.ReadLine();

userlist.Add(user);

Console.WriteLine("Next one? enter key for yes, no for other keys");

next = Console.ReadKey(true).Key == ConsoleKey.Enter ? true : false;

}

userlist.Sort();

Random r = new Random();

List<int> randomList = new List<int>();

for (int i=1; i<=3; i++)

{

int lucky=r.Next(userlist.Count);

if (!randomList.Contains(lucky))

randomList.Add(lucky);

else i--;

}

for (int j=0; j<3; j++)

{

int index = randomList[j];

switch(j)

{

case 0: Console.Write("Congrats first winner, "); break;

case 1: Console.Write("Congrats second winner, "); break;

case 2: Console.Write("Congrats third winner, "); break;

}

Console.Write( userlist[index].ToString());

Console.WriteLine();

}

}

}

**5. Write your own creative C# Console Application.**

**Test 2**

1. Write a C# class named “Distance” that consists of

feet (int), inches(float) and the following member functions:

- A constructor that initialise 0 to the data.

- A constructor that assign data to the object

- showdata( ) that displays the distance object.

- add( ) that adds two distance objects into sum.

- IsTriangle( ) that checks whether given three objects may be the sides of the tringle or not. Return the result by Boolean data type.

Write main program that creates their distance objects, assigned to the data. Find their sum and display. Check whether their sides are the sides of a triangle or not.

using System;

public class Distance

{

private int feet;

private float inches;

public Distance() { feet = 0; inches = 0; }

public Distance(int f, float inch) { feet = f; inches = inch; }

public void Show() {

Console.WriteLine(feet + " feet," + inches +" inches");

}

public void Add(Distance d1, Distance d2)

{

feet = d1.feet + d2.feet;

inches = d1.inches + d2.inches;

if(inches>12)

{

feet++;

inches -= 12;

}

}

public bool IsTriangle(Distance d1, Distance d2, Distance d3)

{

Distance temp = new Distance();

temp.Add(d1, d2);

float twosides = temp.feet \* 12 + temp.inches;

float oneside = d3.feet \* 12 + d3.inches;

if (twosides > oneside)

return true;

else return false;

}

} //Class Distance End

class program

{

static void Main(string[] args)

{

Distance d1 = new Distance(4, 9);

Distance d2 = new Distance(3, 11);

Distance d3=new Distance();

d3.Add(d1, d2);

d3.Show();

Distance d4 = new Distance(5, 9);

Distance d5 = new Distance();

if (d5.IsTriangle(d1,d2,d4) == true) Console.WriteLine("They are sides of triangle");

else Console.WriteLine("They are not sides of triangle");

}

}

2. Write a C# Polymorphic program using Virtual and Override keyword that does the following tasks.

A virtual function Engine() that has basic properties of engine like Power of engine, RPM, no of Cylinder etc. This function should be overridden in child class according to function.

using System;

public class Vehicle

{

private int enginepower;

private int RPM;

private int cylinder;

virtual public void Engine()

{

enginepower = 100;

RPM = 200;

cylinder = 3;

}

}

public class Toyota: Vehicle

{

private int enginepower;

private int RPM;

private int cylinder;

public override void Engine()

{

enginepower = 250;

RPM = 126;

cylinder = 4;

Console.WriteLine("Toyotar Engine data=" + enginepower + "," + RPM + "," + cylinder);

}

}

class program

{

static void Main(string[] args)

{

Vehicle v1 = new Toyota();

v1.Engine();

}

}

3. Create a “Person” class that consists of a name and salary with the following member functions:

getData( ) that reads data from the person object

showData( ) that display data of person object

Write a main( ) program that create n array of Person objects until user user’s enter data. Sort the array in **Ascending** order by salary.

using System;

using System.Collections;

public class Person

{

private string name;

private double salary;

public void getData()

{

Console.Write("Enter your name:");

name = Console.ReadLine();

Console.Write("Enter your salary");

salary = double.Parse(Console.ReadLine());

}

public void showData()

{

Console.WriteLine(name + ", " + salary);

}

public double Salary

{

get { return salary; }

}

public string Name

{

get { return name; }

}

}

class program

{

static void Main(string[] args)

{

ArrayList plist = new ArrayList();

char ch = 'y';

while (ch == 'y')

{

Person p = new Person();

p.getData();

plist.Add(p);

Console.WriteLine("Anymore y/n?");

ch = Console.ReadKey().KeyChar;

}

plist.Sort(new myComparer()); // plist.Reverse(); for descending

Console.WriteLine("\n..::After Ordering::..");

foreach (Person p in plist)

{

Console.WriteLine("Name:{0} Salary:{1}",p.Name,p.Salary );

}

Console.ReadLine();

}

public class myComparer : IComparer

{

int IComparer.Compare(Object xx, Object yy)

{

Person x = (Person)xx;

Person y = (Person)yy;

return x.Salary.CompareTo(y.Salary);

}

}

}

4. Write a C# program that **counts** the objects you have created from main( ) function using the following constructors:

- Static Constructor

- No argument Constructor

using System;

public class Person

{

private static int count;

static Person() { count=0; }

public Person() { count++; }

public int showCount() { return count; }

}

class program

{

static void Main(string[] args)

{

Person p1 = new Person();

Person p2 = new Person();

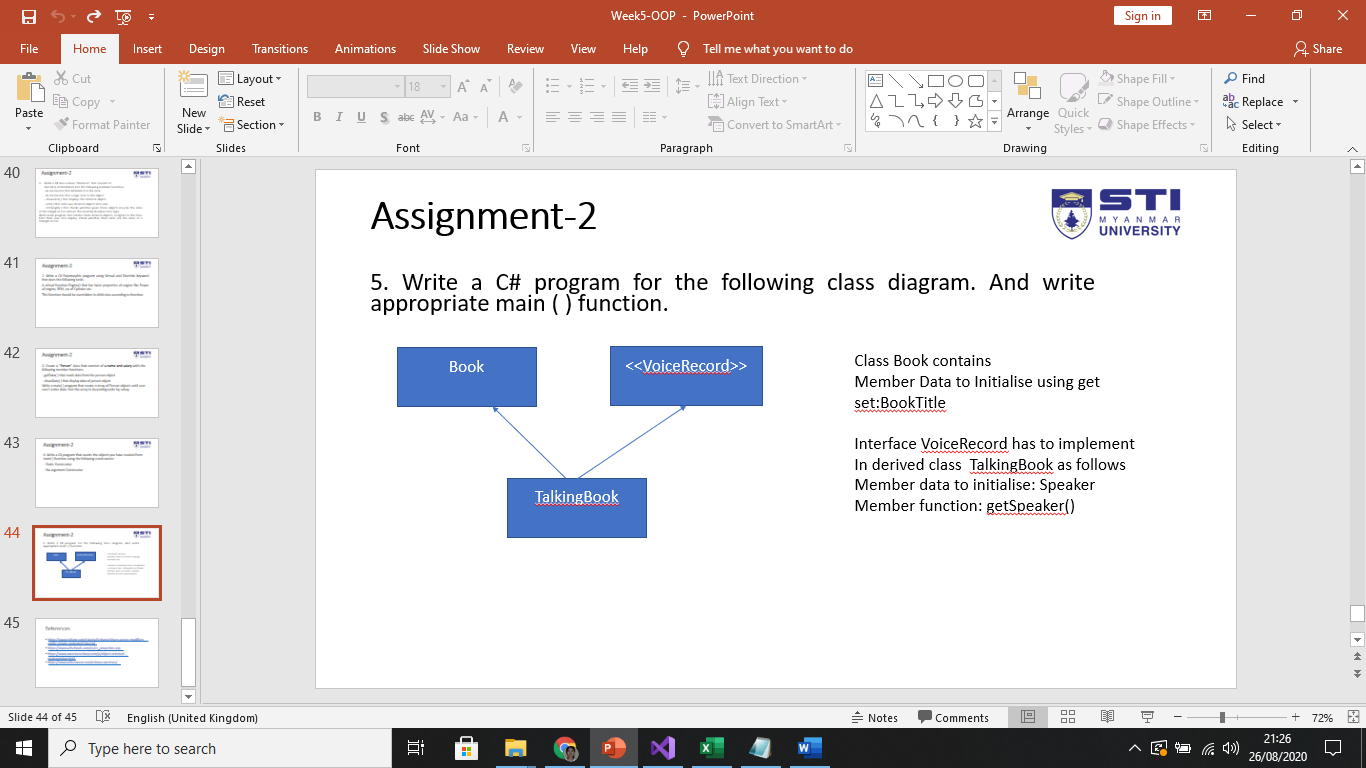
Person p3 = new Person();

Console.WriteLine("Object Count="+ p3.showCount());

}

}

5. Write a C# program for the following class diagram. And write appropriate main ( ) function.



Class Book contains

Member Data to Initialise using get set:BookTitle

Interface VoiceRecord has to implement

In derived class TalkingBook as follows

Member data to initialise: Speaker

Member function: getSpeaker()

using System;

public class Book

{

public string BookTitle { get; set; }

}

public interface VoiceRecord

{

public void getSpeaker();

}

public class takTalkingBook:Book,VoiceRecord

{

string speaker = "Mary";

public void getSpeaker()

{

BookTitle = "Ten Commandments";

Console.WriteLine("Title=" + BookTitle);

Console.WriteLine("Speaker=" + speaker);

}

}

class program

{

static void Main(string[] args)

{

takTalkingBook tk = new takTalkingBook();

tk.getSpeaker();

}

}

**Test3**

1. Write C# Polymorphic classes to perform different tasks as follows;

Base class “Shape” that has

- double PI

- double x, y;

And area( ) function to find the area of an object.

Derived Classes are

**“Rectangle**”, “Circle” and “Sphere”. Then, find the area of the three given objects in the Main ( ) program.

**2. Write a C# File handling program that accepts student’s data and save in the file name “student.txt”. And then show the students data from that file.**

3. Write a C# program that swaps two integer numbers using **pass by reference.**

4. Write a C# **Tollbooth** class that counts the number of cars and trucks for each day. Write Main ( ) function to count the total **cars** and **trucks.**

**5. Write a C# calculator program using Exception handling especially division by zero.**