EAD HOMEWORK\_2

**Q1: Write down the advantages of unary and binary operators**

**Advantages of unary and binary operators:**

unary operators make operations more easier and in less code ,which increases readability ,effectiveness and ease of programmer due to less lines of code

Example: i=i+1🡺 i++ , i=i/7 🡺 i/=7

Where as binary operators use when we perform operation between two operands , it solves every type of binary operation e.g: addition, multiplications, division, etc

**Q2: Suppose a,b,c,d,e,f,g are int type variables and initialized with 0 value. What will be the output of these variables if we apply the given operations. Please attach screenshot of the output and explain the output values procedure clearly.**

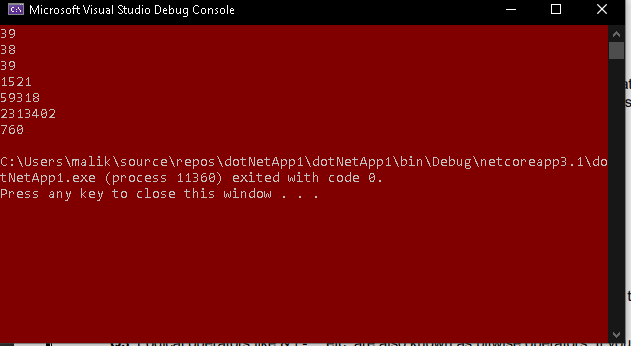
**b = 39; a = ++b c = --b; d = (c++)\*(--a); e = (d++)\*(--c);**

**f = --b + 1; f \*= --e; g = --d / 2;**

**If a becomes a- =50 which variables values will be affected? Please write down their changed values.**

**Answer:**

**Output Values in Alphabetical Sequence ( a,b,c,d,e,f,g ):**



Unary (++i ) , means first increment the value than assaigned for expression evaluation whereas (i++) means first assaign the value the incremented

In above Question, iniatally all variables is equal to zero , then multiple values assaigned ,incremented and decremented , but the main rule of unary

(above said) is same which is the purpose of the question , when we

Dry-run/debug /evaluate the expressions , we get the output(as shown in screenshot)

The other thing in last statement is that --d/2 get 760.5 but due to ‘int’ type and even number (before decimal), it floor the value ofinteger and get 760 as output

🡺If a becomes = -50, then there will be no changing as this value changes at the end so, no affect in overall output

**Q3: Logical operators like &,|,-,^ etc. are also known as bitwise operators. If you want to do AND operation between two numbers like 8(1000) and 5(0101) then its output will be 0.Write a program that takes first argument as the size of array and remaining arguments as numbers from command line arguments. Select two numbers randomly from the arguments numbers and apply bitwise operations using switch statement. Finally display the result in proper output formatting. You can use loop as well to operate your function multiple times.**

using System;

namespace ConsoleApp2

{

class Program

{

static void Main(string[] args)

{

int[] numbers = new int[int.Parse(args[0])];

int size = int.Parse(args[0]);

for (int i = 1; i < args.Length; i++)

{

numbers[i - 1] = int.Parse(args[i]);

}

int number1 = (new Random()).Next(0, size - 1);

int randomNumber1 = numbers[number1];

int number2 = (new Random()).Next(0, size - 1);

int randomNumber2 = numbers[number2];

var bitWiseOperator = "";

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

{

if (num == 1)

{

bitWiseOperator = "|";

}

else if (num == 2)

{

bitWiseOperator = "&";

}

else

{

bitWiseOperator = "^";

}

dynamic output = bitWiseOperator switch

{

"|" => randomNumber1 | randomNumber2,

"&" => randomNumber1 & randomNumber2,

"^" => randomNumber1 ^ randomNumber2,

\_ => "not a bitwiswe operator"

};

Console.WriteLine(

format: "{0} {1} {2} = ",

arg0: randomNumber1,

arg1: bitWiseOperator,

arg2: randomNumber2

);

Console.Write(output);

Console.WriteLine();

Console.WriteLine();

}

}

}

}

**Q4: Write a program that takes size of student subjects and their numbers in command line arguments. Apply Switch expression to display given statements with proper output formatting:**

using System;

namespace ConsoleApp2

{

class Program

{

static void Main(string[] args)

{

int[] a = new int[int.Parse(args[0])];

for (int j = 0; j < args.Length-1; j++)

{

a[j] = int.Parse(args[j+1]);

}

int len = 1;

foreach (int i in a)

{

dynamic gpa = i switch

{

\_ when i < 50 => 0.0,

\_ when i >= 50 && i < 57 => 1.7,

\_ when i >= 57 && i < 64 => 1.7,

\_ when i >= 64 && i < 67 => 2.0,

\_ when i >= 68 && i < 70 => 2.7,

\_ when i >= 70 && i < 75 => 3.0,

\_ when i >= 75 && i < 80 => 3.4,

\_ when i >= 80 && i < 85 => 3.7,

\_ when i >= 85 && i <= 100 => 4.0,

\_ => "incorrect Input"

};

Console.WriteLine($"Your gpa in course {len} is {gpa}");

len++;

}

}

}

}

**Q5: Write a program that takes a floating-point number from user as input and then rounds off the number. Rounding the number should also be based on choice from user. Provide two methods of rounding, One with the banker’s algorithm and the other one with the old method of rounding.**

static void Main(string[] args)

{

Console.WriteLine("Enter Floating Number Which you want to Round off");

string value = Console.ReadLine();

Console.WriteLine("What type of Rounding Algorithm you want ?");

Console.WriteLine("For Banker's Algorithm Enter 1 ");

Console.WriteLine("For Old Algorithm Enter 2 ");

string algo;

int flag = 1;

do

{

algo = Console.ReadLine();

if (int.Parse(algo) == 1 || int.Parse(algo) == 2)

{

flag = 0;

}

else

{

Console.WriteLine("Plz Enter from Given Algorithm");

}

} while (flag == 1);

var algoNum = double.Parse(algo);

var value2 = double.Parse(value);

if (algoNum == 1)

{

// implement bankers logic

Console.WriteLine($"Rounding using Banker's Algorithm will convert {value2} to {System.Convert.ToInt32(value2)}");

}

else if (algoNum == 2)

{

// implement old logic

Console.WriteLine($"Rounding using Standard method will convert {value2} to {Math.Round(value: value2, digits: 0, mode: MidpointRounding.AwayFromZero)}");

}

}

}

**Q6: Differentiate between implicit and explicit type casting with examples. Why explicit type casting is not good?**

**Answer:**

When we assaign a value to large datatype comparatively , it is called implicit type casting. In this casting, no loss of data occurs and automatic type casting takes place

Where as when we assaign value to small datatype, it is called explicit type casting. Here user own do typecasting and here frequently loses data

Explicit type casting is not good because mostly data loss in this casting when small size contain value of large datatype

|  |  |
| --- | --- |
| //implicit Example:  Int a=10;  double b= a; | //Explicit Example  double c = 9.8;  int d =(int)c; |
| Automatic typecasting | User own manual type cast c to int |

**Q7**: Please find out the errors and explain the valid reasons in the snippet if it has.

namespace ConsoleApp2

{

public class Model1

{

private protected int firstVariable;

private int secondVariable;

}

public class Model2 : Model1

{

internal protected int thirdVariable;

public int fourthVariable;

}

public class Model3 : Model2

{

protected int fifthVariable;

}

public class Model4 : Model3

{

public void Driver()

{

Model3 modelObj = new Model3();

modelObj.fifthVariable = 50;//protected member cannot access

fifthVariable = 23;

}

int sixthVariable;

}

class Program

{

static void Main(string[] args)

{

Model2 medel2\_Obj = new Model2();

Model4 model4\_Obj = new Model4();

Model4 model4\_Obj2 = new Model3();

model4\_Obj.secondVariable;

model4\_Obj.fourthVariable;

medel2\_Obj.firstVariable;

}

}

}

namespace ConsoleApp1

{

class Program : Model2

{

static void Main(string[] args)

{

Model1 obj = new Model1(); Program obj\_2 = new Program(); obj\_2.thirdVariable = 1;

obj.firstVariable = 4;

}

}

}

ERRORS:

1. In model4 class (which is derived from model3 class) cannot access or create protected member of model3 (i-e fifthVariable) , but can access public members
2. In main of class program (namespace ConsoleApp2) , there is error

Model4 model4\_Obj2 = new Model3();

This is because child object(Model4) cannot create parent object(Model3). Although parent can create child

Classes

1. In main of class program (namespace ConsoleApp2) , there are errors

model4\_Obj.secondVariable;

model4\_Obj.fourthVariable;

medel2\_Obj.firstVariable;

these errors is due to access private or public members of parents or grand parent

you cannot access private or protected member of parent class in main program in same assembly and namespace too.

Note: You can access public member of parent class in main

4)In namespace ConsoleApp1 there is inheritance of Model2 class of namespace ConsoleApp2 , which is not accessible. (class Program : Model2)

You cannot inherit class of other namespace, only can inherit in same namespace

5)Similarly errors in main of namespace ConsoleApp1.

These errors are due to access other namespace’s classes and protected as private members of parent class which is not accessible and also not possible in real world too.

The table given below shows access modifiers :

