Parte A

Ejemplo 1 de video

```
using System;
using System.Collections.Generic ; using System.Linq ;
using System.Text;
using System.Threading.Tasks ;
 namespace Operadores_aritmeticos
 {
class Program
  {
static void Main ( string [ ] args )
{
// Operadores aritmeticos
int num = 6 , num2 = 5 ;
}
Console.WriteLine (" El resultado de la suma es: + ( num - num2 ) ) ;
Console.ReadKey();
     }
}
```

Ejemplo 2 de video

```
using System ;
using System.Collections.Generic;
 using System.Linq ;
using System.Text ;
using System.Threading.Tasks ;
 namespace Operadores_aritmeticos
{

class Program
   {
     static void Main ( string [ ] args )
      {
// Operadores aritmeticos
double num , pot , resultado ;
Console.WriteLine (" Digite el numero que quiere elevar :" ) ;
num = Convert.ToDouble ( Console.ReadLine ( ) ) ;
Console.WriteLine ( " Digite a la potencia que quiere elevar : " ) ;
 pot = Convert.ToDouble ( Console.ReadLine ( ) ) ;
```

resultado Math,Pow (num , pot));
Console.WriteLine (" El resultado es : + resultado") ;
Console.ReadKey ( ) ;}}}

Ejemplo 1 buscado

```
Private m_longBase As Double
    Private m_shortBase As Double
    Private m_leftLeg As Double
    Private m_rightLeg As Double

    Public Sub New(ByVal longbase As Double, ByVal shortbase As Double,
ByVal leftLeg As Double, ByVal rightLeg As Double)
        m_longBase = Math.Abs(longbase)
        m_shortBase = Math.Abs(shortbase)
        m_leftLeg = Math.Abs(leftLeg)
        m_rightLeg = Math.Abs(rightLeg)
    End Sub

    Private Function GetRightSmallBase() As Double
        GetRightSmallBase = (Math.Pow(m_rightLeg, 2) - Math.Pow(m_leftLeg,
2) + Math.Pow(m_longBase, 2) + Math.Pow(m_shortBase, 2) - 2 * m_shortBase *
m_longBase) / (2 * (m_longBase - m_shortBase))
    End Function

    Public Function GetHeight() As Double
        Dim x As Double = GetRightSmallBase()
        GetHeight = Math.Sqrt(Math.Pow(m_rightLeg, 2) - Math.Pow(x, 2))
    End Function

    Public Function GetSquare() As Double
        GetSquare = GetHeight() * m_longBase / 2
    End Function

    Public Function GetLeftBaseRadianAngle() As Double
        Dim sinX As Double = GetHeight() / m_leftLeg
        GetLeftBaseRadianAngle = Math.Round(Math.Asin(sinX), 2)
    End Function

    Public Function GetRightBaseRadianAngle() As Double
        Dim x As Double = GetRightSmallBase()
        Dim cosX As Double = (Math.Pow(m_rightLeg, 2) + Math.Pow(x, 2) -
Math.Pow(GetHeight(), 2)) / (2 * x * m_rightLeg)
        GetRightBaseRadianAngle = Math.Round(Math.Acos(cosX), 2)
    End Function

    Public Function GetLeftBaseDegreeAngle() As Double
        Dim x As Double = GetLeftBaseRadianAngle() * 180 / Math.PI
        GetLeftBaseDegreeAngle = Math.Round(x, 2)
    End Function
```

```
    Public Function GetRightBaseDegreeAngle() As Double
        Dim x As Double = GetRightBaseRadianAngle() * 180 / Math.PI
        GetRightBaseDegreeAngle = Math.Round(x, 2)
    End Function

    Public Shared Sub Main()
        Dim trpz As MathTrapezoidSample = New MathTrapezoidSample(20, 10, 8,
6)
        Console.WriteLine("The trapezoid's bases are 20.0 and 10.0, the
trapezoid's legs are 8.0 and 6.0")
        Dim h As Double = trpz.GetHeight()
        Console.WriteLine("Trapezoid height is: " + h.ToString())
        Dim dxR As Double = trpz.GetLeftBaseRadianAngle()
        Console.WriteLine("Trapezoid left base angle is: " + dxR.ToString()
+ " Radians")
        Dim dyR As Double = trpz.GetRightBaseRadianAngle()
        Console.WriteLine("Trapezoid right base angle is: " + dyR.ToString()
+ " Radians")
        Dim dxD As Double = trpz.GetLeftBaseDegreeAngle()
        Console.WriteLine("Trapezoid left base angle is: " + dxD.ToString()
+ " Degrees")
        Dim dyD As Double = trpz.GetRightBaseDegreeAngle()
        Console.WriteLine("Trapezoid left base angle is: " + dyD.ToString()
+ " Degrees")
    End Sub
End Class
```

Ejemplo 2 buscado

```
/// <summary>
/// The following class represents simple functionality of the trapezoid.
/// </summary>
using namespace System;

public ref class MathTrapezoidSample
{
private:
   double m_longBase;
   double m_shortBase;
   double m_leftLeg;
   double m_rightLeg;

public:
   MathTrapezoidSample( double longbase, double shortbase, double leftLeg,
double rightLeg )
   {
      m_longBase = Math::Abs( longbase );
      m_shortBase = Math::Abs( shortbase );
      m_leftLeg = Math::Abs( leftLeg );
      m_rightLeg = Math::Abs( rightLeg );
   }
```

```
private:
   double GetRightSmallBase()
   {
      return (Math::Pow( m_rightLeg, 2.0 ) - Math::Pow( m_leftLeg, 2.0 ) +
Math::Pow( m_longBase, 2.0 ) + Math::Pow( m_shortBase, 2.0 ) - 2 *
m_shortBase * m_longBase) / (2 * (m_longBase - m_shortBase));
   }


public:
   double GetHeight()
   {
      double x = GetRightSmallBase();
      return Math::Sqrt( Math::Pow( m_rightLeg, 2.0 ) - Math::Pow( x, 2.0 )
);
   }

   double GetSquare()
   {
      return GetHeight() * m_longBase / 2.0;
   }

   double GetLeftBaseRadianAngle()
   {
      double sinX = GetHeight() / m_leftLeg;
      return Math::Round( Math::Asin( sinX ), 2 );
   }

   double GetRightBaseRadianAngle()
   {
      double x = GetRightSmallBase();
      double cosX = (Math::Pow( m_rightLeg, 2.0 ) + Math::Pow( x, 2.0 ) -
Math::Pow( GetHeight(), 2.0 )) / (2 * x * m_rightLeg);
      return Math::Round( Math::Acos( cosX ), 2 );
   }

   double GetLeftBaseDegreeAngle()
   {
      double x = GetLeftBaseRadianAngle() * 180 / Math::PI;
      return Math::Round( x, 2 );
   }

   double GetRightBaseDegreeAngle()
   {
      double x = GetRightBaseRadianAngle() * 180 / Math::PI;
      return Math::Round( x, 2 );
   }

};
```

```
int main()
{
   MathTrapezoidSample^ trpz = gcnew MathTrapezoidSample( 20.0,10.0,8.0,6.0
);
   Console::WriteLine( "The trapezoid's bases are 20.0 and 10.0, the
trapezoid's legs are 8.0 and 6.0" );
   double h = trpz->GetHeight();
   Console::WriteLine( "Trapezoid height is: {0}", h.ToString() );
   double dxR = trpz->GetLeftBaseRadianAngle();
   Console::WriteLine( "Trapezoid left base angle is: {0} Radians",
dxR.ToString() );
   double dyR = trpz->GetRightBaseRadianAngle();
   Console::WriteLine( "Trapezoid right base angle is: {0} Radians",
dyR.ToString() );
   double dxD = trpz->GetLeftBaseDegreeAngle();
   Console::WriteLine( "Trapezoid left base angle is: {0} Degrees",
dxD.ToString() );
   double dyD = trpz->GetRightBaseDegreeAngle();
   Console::WriteLine( "Trapezoid left base angle is: {0} Degrees",
dyD.ToString() );
}
```

Parte B

```
using System ;
using System.Collections.Generic;
using System.Linq ;
using System.Text ;
using System.Threading.Tasks ;

using System.Collections.Generic ;
namespace Operadores
{
class Program
 {
static void Main ( string [ ] args )
 {
// Operadores relaciones
double peso ;
Console.WriteLine ( " Digita tu peso : " ) ;
peso = Convert.ToDouble ( Console.ReadLine ( ) ) ;
if (peso > 100){
Console.WriteLine("tu peso es normal");
 }
Console.ReadKey();
     }
  }
}
```