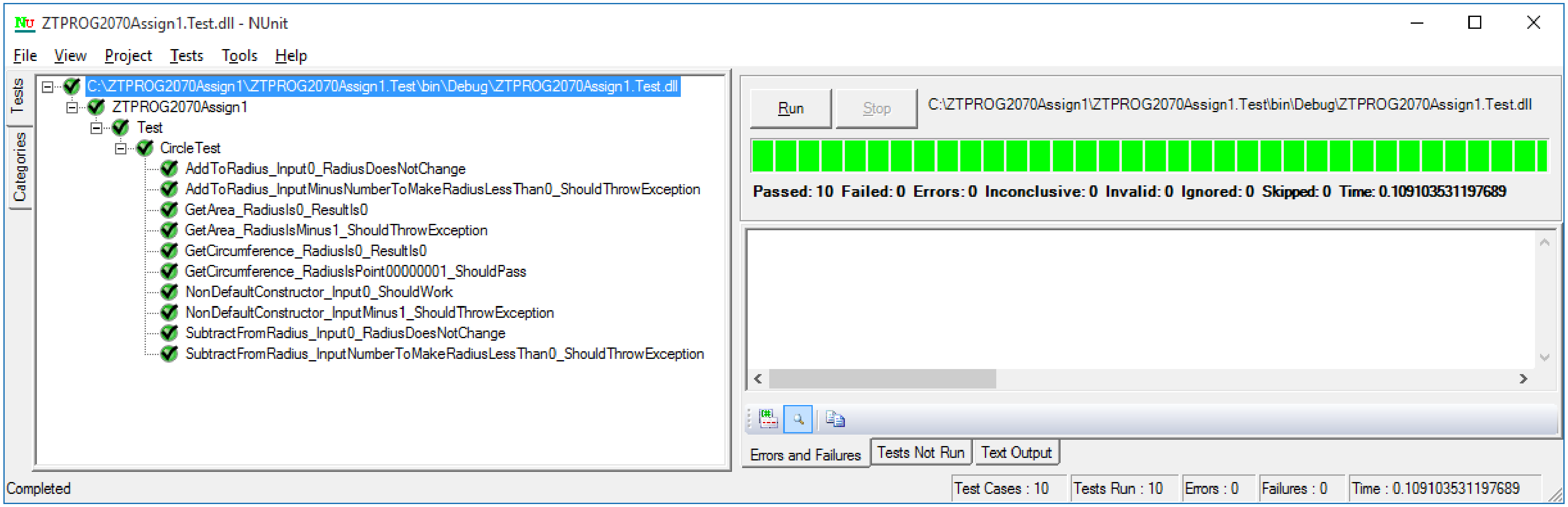

```
C:\ZTPROG2070Assign1\ZTPROG2070Assign1.Test\bin\Debug\ZTPROG2070Assign1.Test.dll
  ZTPROG2070Assign1
    Test
      CircleTest
        AddToRadius_Input0_RadiusDoesNotChange
        AddToRadius_InputMinusNumberToMakeRadiusLessThan0_ShouldThrowException
        GetArea_RadiusIs0_ResultIs0
        GetArea_RadiusIsMinus1_ShouldThrowException
        GetCircumference_RadiusIs0_ResultIs0
        GetCircumference_RadiusIsPoint00000001_ShouldPass
        NonDefaultConstructor_Input0_ShouldWork
        NonDefaultConstructor_InputMinus1_ShouldThrowException
        SubtractFromRadius_Input0_RadiusDoesNotChange
        SubtractFromRadius_InputNumberToMakeRadiusLessThan0_ShouldThrowException
```

C:\ZTPROG2070Assign1\ZTPROG2070Assign1.Test\bin\Debug\ZTPROG2070Assign1.Test.dll

Passed: 10 Failed: 0 Errors: 0 Inconclusive: 0 Invalid: 0 Ignored: 0 Skipped: 0 Time: 0.109103531197689

Completed | Test Cases : 10 | Tests Run : 10 | Errors : 0 | Failures : 0 | Time : 0.109103531197689

Unit Tests for Non-default Constructor:

1. public void NonDefaultConstructor_Input0_ShouldWork()
   Input 0 : for 0 is a special number in all numbers

2. public void NonDefaultConstructor_InputMinus1_ShouldThrowException ()
   Input -1 : for the radius of the circle cannot be less than 0

Unit Tests for AddToRadius():

1. public void AddToRadius_Input0_RadiusDoesNotChange()
   Input 0 : for 0 is a special number in all numbers
   Radius 99999999.99 : a relatively large double to confirm a large number works

2. public void AddToRadius_InputMinusNumberToMakeRadiusLessThan0_ShouldThrowException()
   Input -999999999 : a relatively small minus number added to radius to make exception
   Radius 1 : for 1 is a special number that is the smallest positive integer

Unit Tests for SubtractFromRadius():

1. public void SubtractFromRadius_Input0_RadiusDoesNotChange ()
   Input 0 : for 0 is a special number in all numbers
   Radius 0.00000001 : a relatively small double to confirm a small number works

2. public void SubtractFromRadius_InputNumberToMakeRadiusLessThan0_ShouldThrowException ()
   Input 0.00000002 : a relatively small double to confirm a small number works, subtracting from radius to make radius a little less than zero, to throw an exception
   Radius 0.00000001 : a relatively small double to confirm a small number works

Unit Tests for GetCircumference ():

1. public void GetCircumference_RadiusIs0_ResultIs0 ()
   Input 0 : for 0 is a special number in all numbers

2. public void GetCircumference_RadiusIsPoint00000001_ShouldPass ()
   Input 0.00000001d : a relatively small number to confirm a small number works

Unit Tests for GetArea ():

1. public void GetArea_RadiusIs0_ResultIs0 ()
   Input 0 : for 0 is a special number in all numbers

2. public void GetArea_RadiusIsMinus1_ShouldThrowException ()
   Input -1 : for radius is a public variable in Circle class, it can be changed to an invalid value less than 0 by directly signing like “circle.radius = -1”