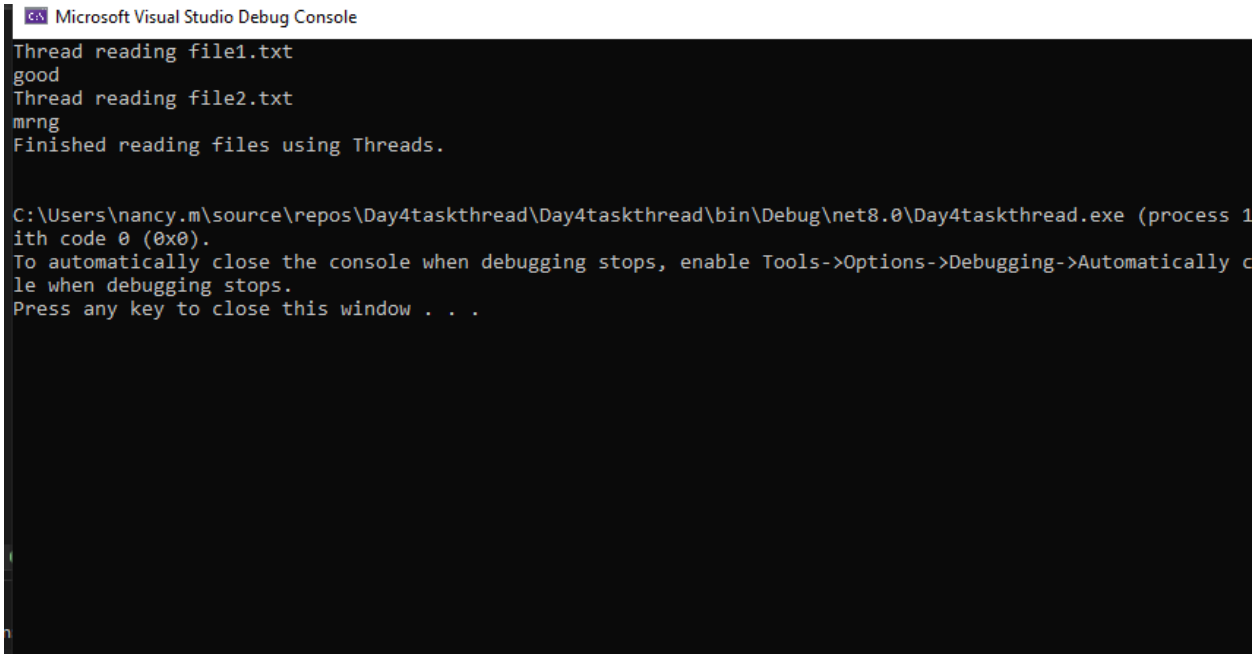
DAY 4

DATE:07-08-25
NAME: NANCY M

1. Create two threads to read two separate text files

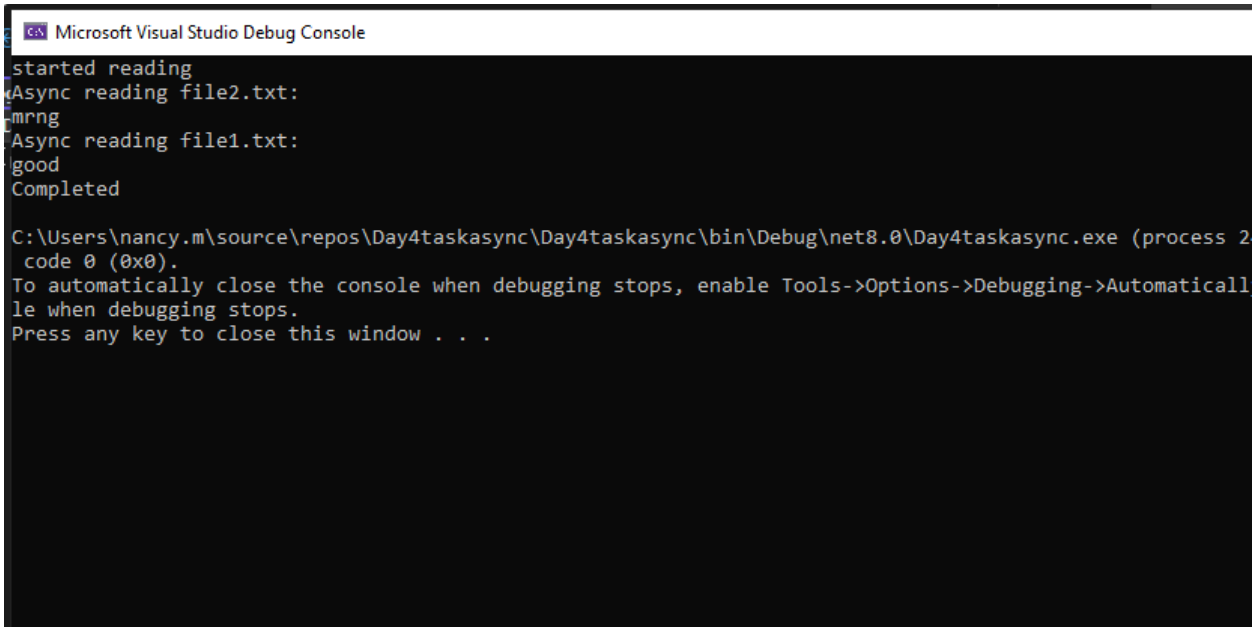
```
namespace Day4taskthread;
class Program
{
   static void Main()
   {
      Thread f1 = new Thread(() =>
ReadFile("C:\\Users\\nancy.m\\source\\repos\\file1.txt"));
      Thread f2 = new Thread(() =>
ReadFile("C:\\Users\\nancy.m\\source\\repos\\file2.txt"));
      f1.Start();
      f2.Start();

      f1.Join();
      f2.Join();

      Console.WriteLine("Finished reading files using Threads.\n");
   }

   static void ReadFile(string fileName)
   {
      string content = File.ReadAllText(fileName);
      string shortName = Path.GetFileName(fileName);
      Console.WriteLine($"Thread reading {shortName}");
      Console.WriteLine(content);
   }
}
```



1.1 Same two files, try to read using the Task Async await.

```
namespace Day4taskasync;
class Program
{
    static async Task Main()
    {
        Console.WriteLine("started reading");

        Task f1 = ReadFileAsync("C:\\Users\\nancy.m\\source\\repos\\file1.txt");
        Task f2 = ReadFileAsync("C:\\Users\\nancy.m\\source\\repos\\file2.txt");

        await Task.WhenAll(f1, f2);

        Console.WriteLine("Completed");
    }

    static async Task ReadFileAsync(string fileName)
    {
        string content = await File.ReadAllTextAsync(fileName);
        string shortName = Path.GetFileName(fileName);
        Console.WriteLine($"Async reading {shortName}:");
        Console.WriteLine(content);
```

```
    }
}
```



2. (Optional) Create delegate use case between teacher class and student class:

- teacher method should have test_completed() method passed as delegate to student

- student class should have a method write_test() which will inturn call the parent delegate.

```
using System;
namespace Day4Delegate
{
    class Program
    {
        public delegate void TestDelegate();

        static void Main(string[] args)
        {
            Teacher teacher = new Teacher();
            Student student = new Student();
            student.WriteTest(teacher.TestCompletednotify);
        }
```

```
    }
    class Teacher
    {
        public void TestCompletednotify()
        {
            Console.WriteLine("Teacher: Student has completed the test.");
        }
    }
    class Student
    {
        public void WriteTest(Program.TestDelegate noticeteacher)
        {
            Console.WriteLine(" Writing the test");
            Console.WriteLine(" Test completed.");
            noticeteacher();
        }
    }
}
```