Tuesday, February 8, 2022

C# EXCEPTION HANDLING

Topics: try, catch, finally.

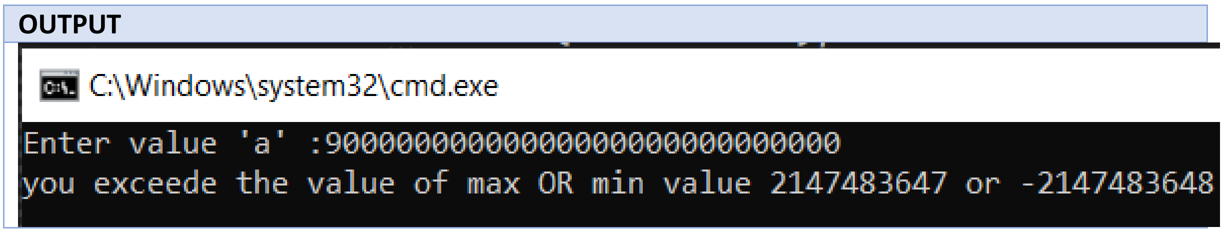
NB_Healthcare.Tech
C# Assignment 12

Q. Write a simple division program and handle three exceptions discussed in the class., also add super exception at the last.

CODE

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

namespace Day_12project1
{
    internal class Program
    {
        static void Main(string[] args)
        {

/******************************************************************************
*******************
             *  AUTHOR : N Prudhvi
*
             *  PURPOSE:Write a simple division program and handle three
exceptions discussed in the class., *
             *          also add super exception at the last
*
             *
*******************************************************************************
****************/
            try
            {
                Console.Write("Enter value 'a' :");
                int a = Convert.ToInt32(Console.ReadLine());
                Console.WriteLine();
                Console.Write("Enter value 'b' :");
                int b = Convert.ToInt32(Console.ReadLine());
                int c = a / b;
                Console.WriteLine();
                Console.WriteLine("Answer ="+c);
            }
            catch (OverflowException ex)
            {
                Console.WriteLine($"you exceede the value of max value
{int.MaxValue} or {int.MinValue}");
            }
            catch(FormatException ex)
            {
                Console.WriteLine($"give the valid format");
            }
            catch (DivideByZeroException ex)
            {
                Console.WriteLine("You can not divide with zero");
            }
            catch(Exception ex)
            {
                Console.WriteLine("The following error can not be done pls
contact the devp : Mr_x@alpabet.in");
            }

        }
    }
}
```

OUTPUT



```
Enter value 'a' :9000000000000000000000000000
you exceede the value of max OR min value 2147483647 or -2147483648
```

Q. What is the use of "finally" block illustrate with an example.

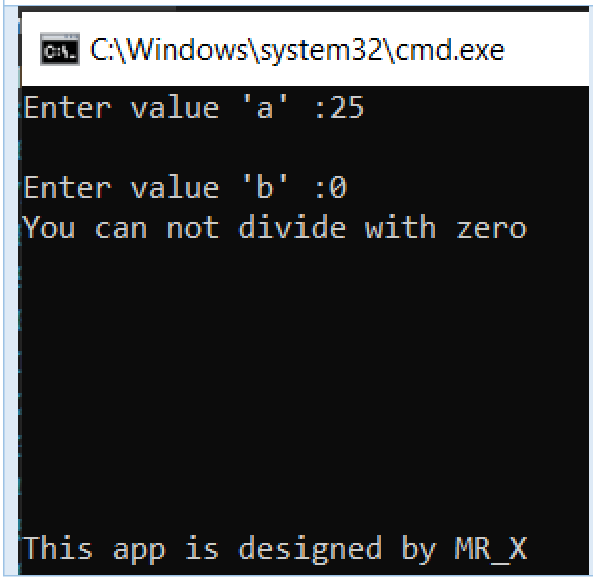
A. **Finally block :**

CODE

```
using System;

namespace Day_12project1
{
    internal class Program
    {
        static void Main(string[] args)
        {

/*******************************************************************************
  * AUTHOR : N Prudhvi                                                         *
  * PURPOSE: Write a simple division program and handle three exceptions      *
  *          discussed in the class., also add super exception at the last    *
  ****************************************************************************/
         try
         {
                Console.Write("Enter value 'a' :");
                int a = Convert.ToInt32(Console.ReadLine());
                Console.WriteLine();
                Console.Write("Enter value 'b' :");
                int b = Convert.ToInt32(Console.ReadLine());
                int c = a / b;
                Console.WriteLine();
                Console.WriteLine("Answer ="+c);
            }
            catch (OverflowException ex)
            {
                Console.WriteLine($"you exceede the value of max value
{int.MaxValue} or {int.MinValue}");
            }
            catch(FormatException ex)
            {
                Console.WriteLine($"give the valid format");
            }
            catch (DivideByZeroException ex)
            {
                Console.WriteLine("You can not divide with zero");
            }
            catch(Exception ex)
            {
                Console.WriteLine("The following error can not be done pls
contact the devp : Mr_x@alpabet.in");
            }
            finally
            {
                Console.WriteLine("\n\n\n\n\n\n\nThis app is designed by
MR_X");
```

```
            }
        }
    }
}
```

OUTPUT



```
Enter value 'a' :25

Enter value 'b' :0
You can not divide with zero





This app is designed by MR_X
```

Q. Research and write at least 6 exceptions that occur in C# with sample code.

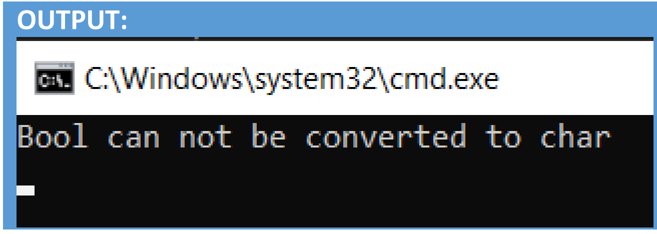
1 Invalid Cast Exception

Reason: An Invalid Cast Exception is thrown when the conversion of an instance of one type to another type is not supported.

CODE:

```
internal class Program
    {
        static void Main(string[] args)
        {
            try
            {
                bool a = true;
                char c = Convert.ToChar(a);
            }
            catch (InvalidCastException ex)
            {
                Console.WriteLine("Bool can not be converted to char");
            }
            Console.ReadLine();
        }
    }
```

OUTPUT:



```
Bool can not be converted to char
```

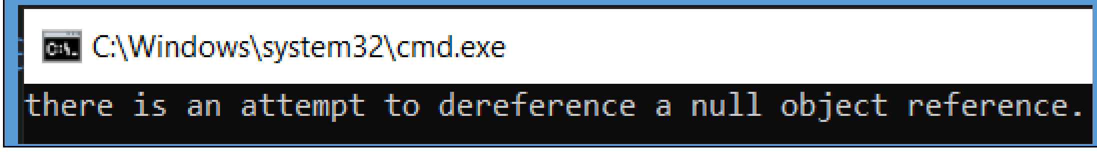
2. Null Reference Exception

Reason: A Null Reference Exception is thrown when you try to access a member on a type whose value is null.

CODE:

```
static void Main(string[] args)
        {
            try
            {
                object o = null;
                object a = o.ToString();
            }
            catch (NullReferenceException ex)
            {
                Console.WriteLine("there is an attempt to dereference a null 
object reference.");
            }


        }
```

OUTPUT:



```
there is an attempt to dereference a null object reference.
```

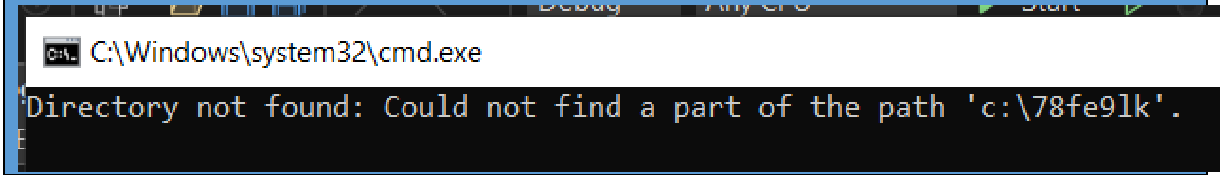
3 Directory Not Found Exception

Reason:

CODE:

```
static void Main(string[] args)
  {
    try
    {
      string dir = @"c:\78fe9lk"
      Directory.SetCurrentDirectory(dir);
    }
    catch (DirectoryNotFoundException dirEx)
    {
    Console.WriteLine("Directory not found " +);
    }
  }
```

OUTPUT :



```
Directory not found: Could not find a part of the path 'c:\78fe9lk'.
```

4 OUT OF MEMORY EXCEPTION
Reason: There is a lack of contiguous memory for the allocations required
CODE:

Q. What is Exception Handling and why we need exception handling

A. Exception handling is used in to define user defined expectations. Rather throwing technical stuff on the console which may sound like verbose! and tends to feel like an error to the user, putting some graceful massage and knowledge user with mistake what he is doing which makes a sense for him, to understand where he is going wrong. So for such scenarios we require exception handling.

Q. 5 points about exception handling discussed in class.

A.

- **Exception handling is done to handle the errors for the user to send a graceful massage to the user to knowledge the user what the mistake he is doing.**
- **Single block “try” can have multiple catches blocks.**
- **Super exception should be written at the last.**
- **Statements written at the finally block will be executed regardless whether exception occurs.**
- **Syntax of exception handling: try, catch, and finally.**

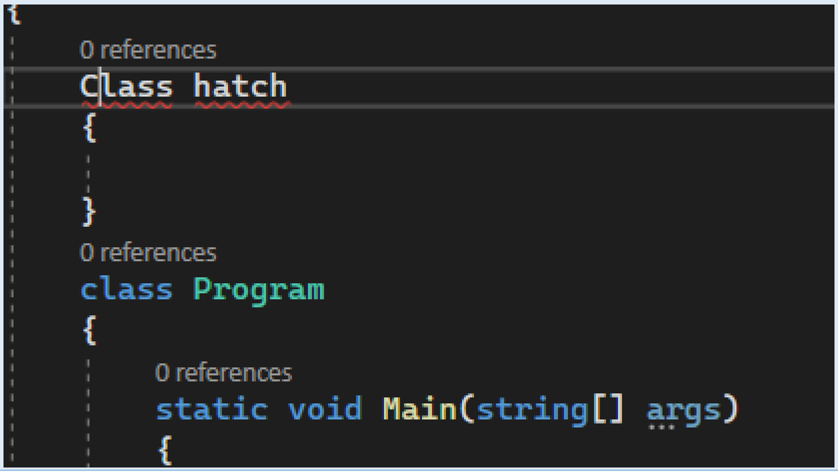
Q. What is compilation and Runtime error Write at least 3 differences between them

Compile errors	Runtime errors
The compiler detects these syntax errors.	These errors are not detected by the compiler. And produce wrong results
They prevent the code from running because it detects some syntax errors.	Runtime errors will not allow to execute the code
The report contains syntax errors such as missing semicolons(;), misspelled keywords and identifiers, etc.	Errors such as dividing a number by zero, finding the square root of a negative number, etc. are included.

Q. 6 snippets of compilation error

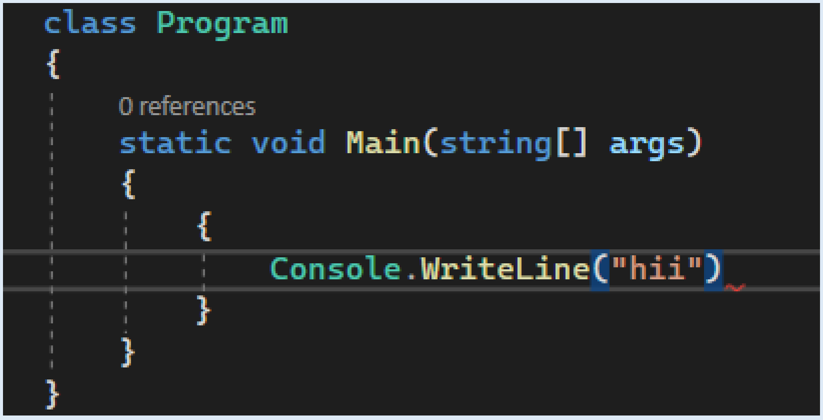
A.

Case sensitive



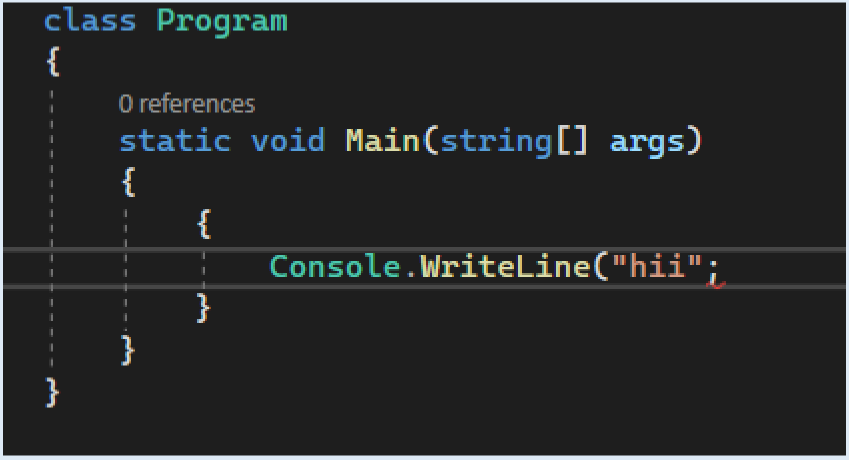
```
{
    0 references
    Class hatch
    {

    }
    0 references
    class Program
    {
        0 references
        static void Main(string[] args)
        {
```

Missing semi colon.



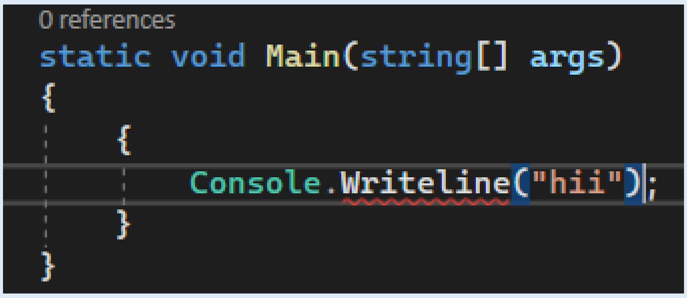
```
class Program
{
    0 references
    static void Main(string[] args)
    {
        {
            Console.WriteLine("hii")
        }
    }
}
```

Semi colon missing



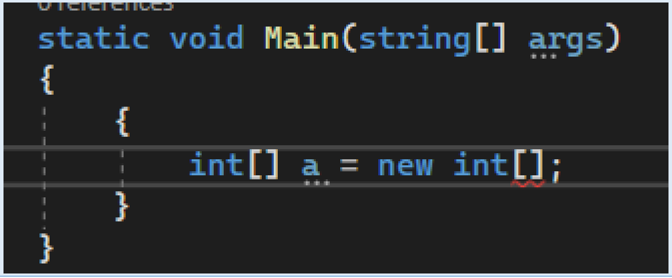
```
class Program
{
    0 references
    static void Main(string[] args)
    {
        {
            Console.WriteLine("hii";
        }
    }
}
```

Case sensitive



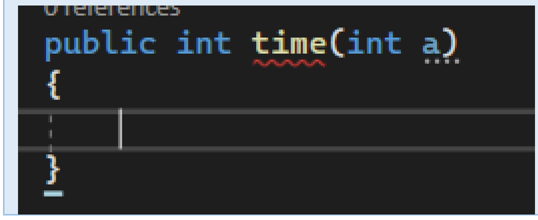
```
0 references
static void Main(string[] args)
{
    {
        Console.Writeline("hii");
    }
}
```

Mention the size of array



```
0 references
static void Main(string[] args)
{
    {
        int[] a = new int[];
    }
}
```

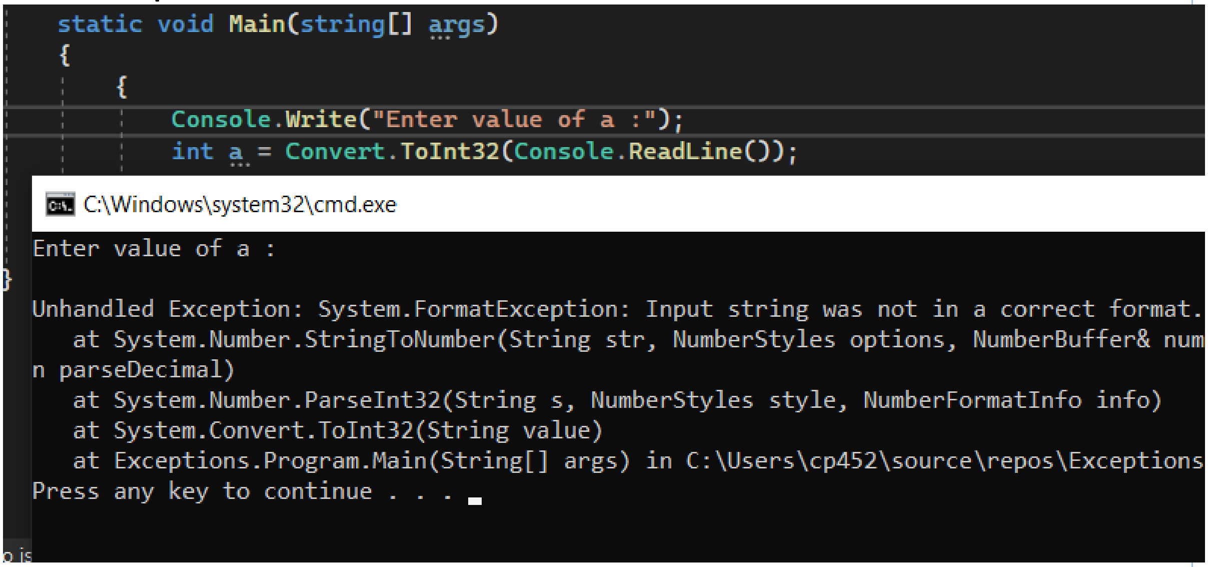
Should use return.



```
0 references
public int time(int a)
{
    |
}
```

Q. 6 runtime errors snippets

Format exception

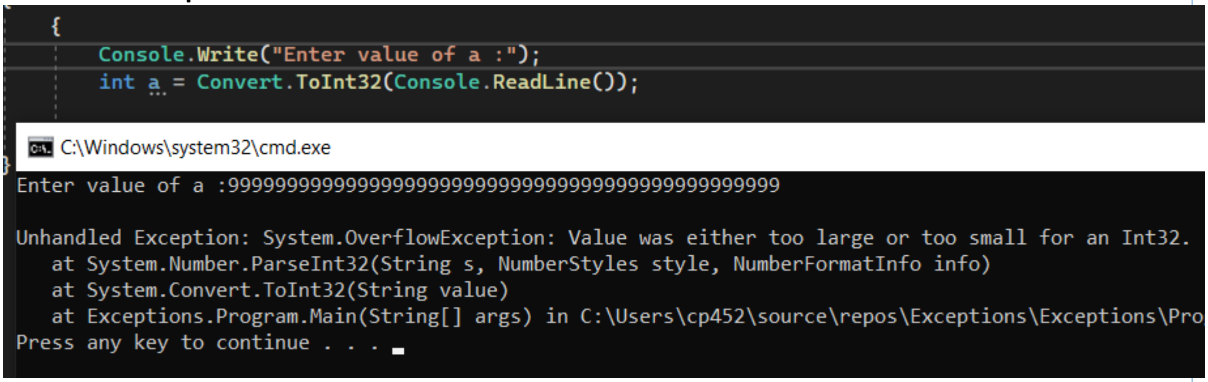


```
static void Main(string[] args)
{
    {
        Console.Write("Enter value of a :");
        int a = Convert.ToInt32(Console.ReadLine());
```

C:\Windows\system32\cmd.exe

```
Enter value of a :

Unhandled Exception: System.FormatException: Input string was not in a correct format.
   at System.Number.StringToNumber(String str, NumberStyles options, NumberBuffer& num
n parseDecimal)
   at System.Number.ParseInt32(String s, NumberStyles style, NumberFormatInfo info)
   at System.Convert.ToInt32(String value)
   at Exceptions.Program.Main(String[] args) in C:\Users\cp452\source\repos\Exceptions
Press any key to continue . . .
```

Over flow exception

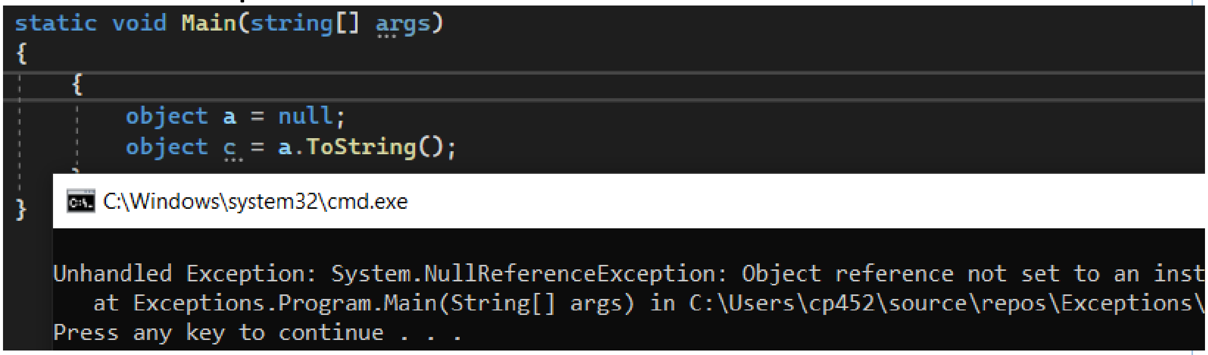


```
{
    Console.Write("Enter value of a :");
    int a = Convert.ToInt32(Console.ReadLine());
```

C:\Windows\system32\cmd.exe

```
Enter value of a :99999999999999999999999999999999999999999999999

Unhandled Exception: System.OverflowException: Value was either too large or too small for an Int32.
   at System.Number.ParseInt32(String s, NumberStyles style, NumberFormatInfo info)
   at System.Convert.ToInt32(String value)
   at Exceptions.Program.Main(String[] args) in C:\Users\cp452\source\repos\Exceptions\Exceptions\Pro
Press any key to continue . . .
```

Null reference exception

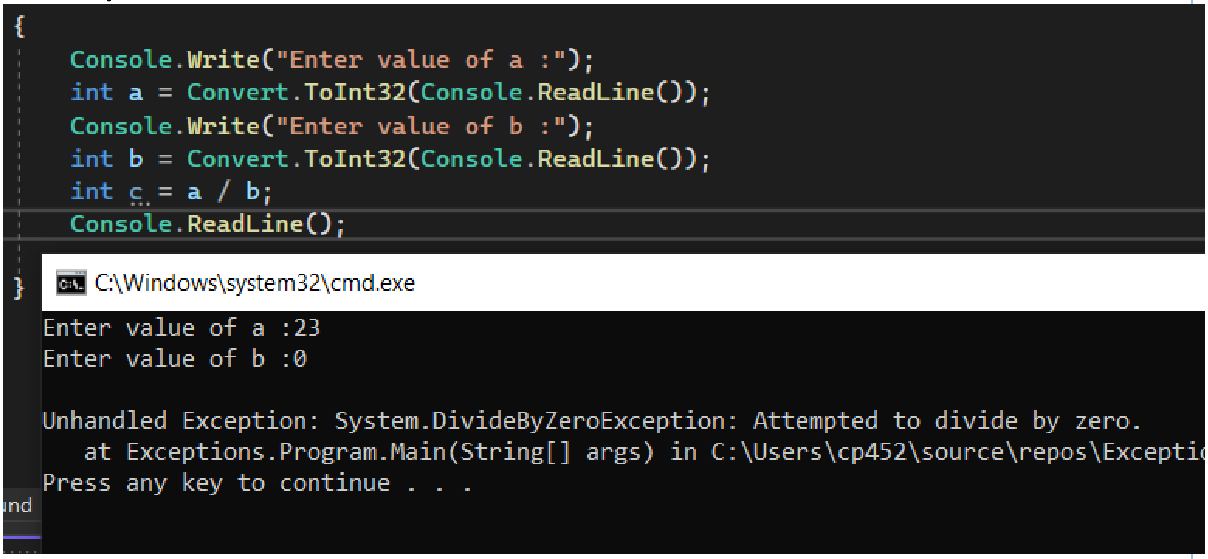


```
static void Main(string[] args)
{
    {
        object a = null;
        object c = a.ToString();
}
```

C:\Windows\system32\cmd.exe

```
Unhandled Exception: System.NullReferenceException: Object reference not set to an inst
   at Exceptions.Program.Main(String[] args) in C:\Users\cp452\source\repos\Exceptions\
Press any key to continue . . .
```

Divide by zero

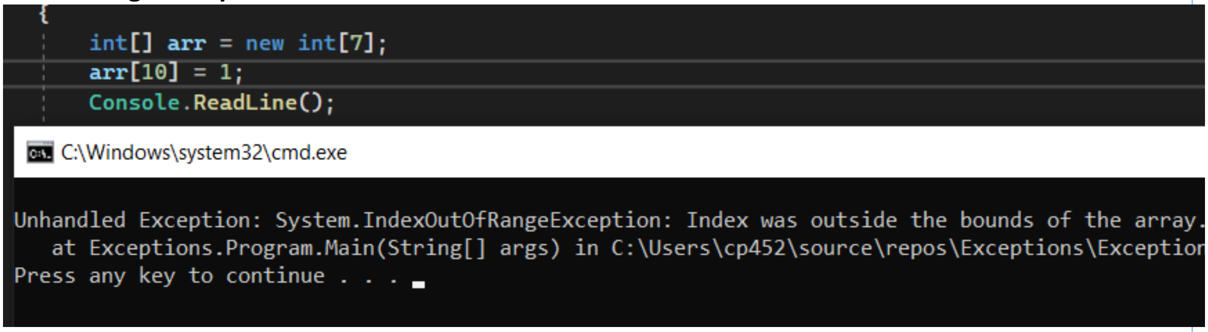


```
{
    Console.Write("Enter value of a :");
    int a = Convert.ToInt32(Console.ReadLine());
    Console.Write("Enter value of b :");
    int b = Convert.ToInt32(Console.ReadLine());
    int c = a / b;
    Console.ReadLine();

}
```

C:\Windows\system32\cmd.exe

```
Enter value of a :23
Enter value of b :0

Unhandled Exception: System.DivideByZeroException: Attempted to divide by zero.
   at Exceptions.Program.Main(String[] args) in C:\Users\cp452\source\repos\Excepti
Press any key to continue . . .
```

Out of range exception

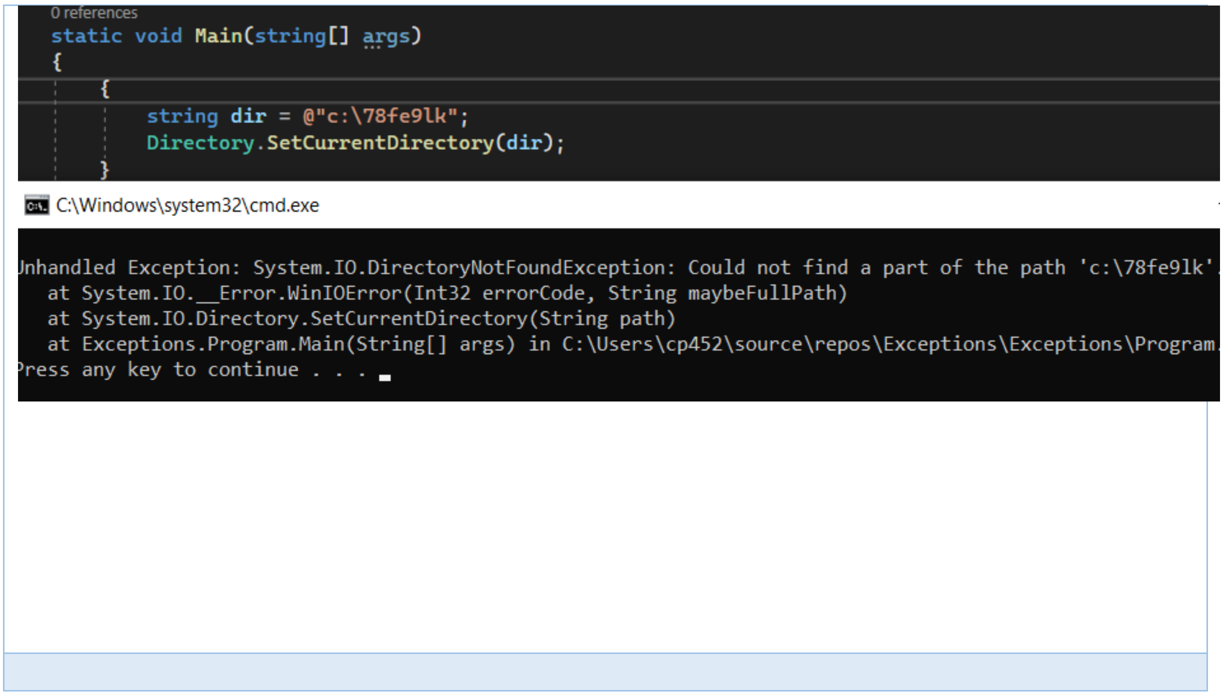


```
{
    int[] arr = new int[7];
    arr[10] = 1;
    Console.ReadLine();
```

C:\Windows\system32\cmd.exe

```
Unhandled Exception: System.IndexOutOfRangeException: Index was outside the bounds of the array.
   at Exceptions.Program.Main(String[] args) in C:\Users\cp452\source\repos\Exceptions\Exception
Press any key to continue . . .
```

Directory not found exception



```
0 references
static void Main(string[] args)
{
    {
        string dir = @"c:\78fe9lk";
        Directory.SetCurrentDirectory(dir);
    }
```

C:\Windows\system32\cmd.exe

```
Unhandled Exception: System.IO.DirectoryNotFoundException: Could not find a part of the path 'c:\78fe9lk'.
   at System.IO.__Error.WinIOError(Int32 errorCode, String maybeFullPath)
   at System.IO.Directory.SetCurrentDirectory(String path)
   at Exceptions.Program.Main(String[] args) in C:\Users\cp452\source\repos\Exceptions\Exceptions\Program.
Press any key to continue . . .
```

-Thank you.