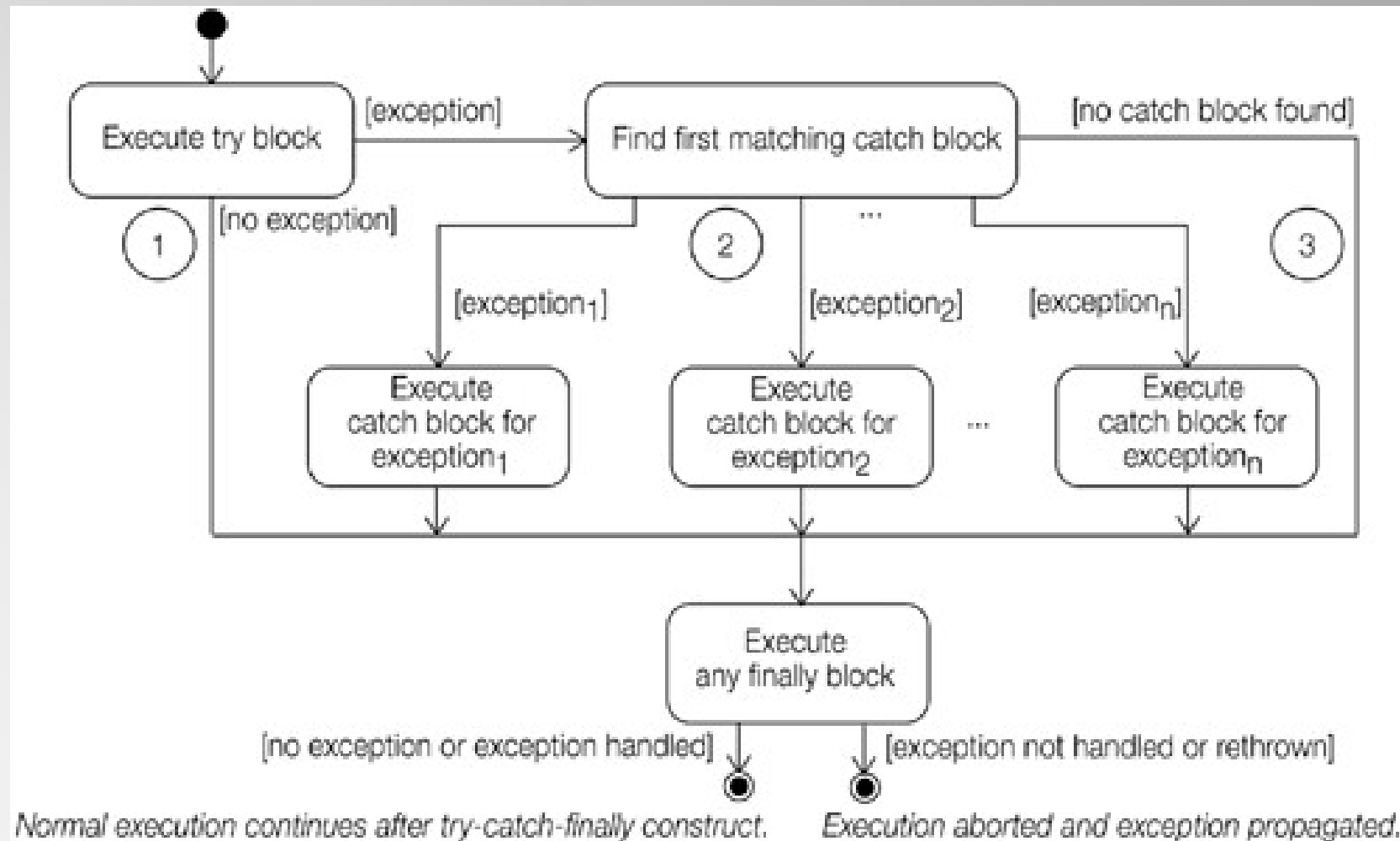


# try catch finally - Ablaufdarstellung



# try catch finally - Beispiel



```
public class EHClass
{
    void ReadFile(int index)
    {
        // To run this code, substitute a valid path from your local machine
        string path = @"c:\users\public\test.txt";
        System.IO.StreamReader file = new System.IO.StreamReader(path);
        char[] buffer = new char[10];
        try
        {
            file.ReadBlock(buffer, index, buffer.Length);
        }
        catch (System.IO.IOException e)
        {
            Console.WriteLine("Error reading from {0}. Message = {1}", path, e.Message);
        }

        finally
        {
            if (file != null)
            {
                file.Close();
            }
        }
        // Do something with buffer...
    }
}
```

Ist im Stack-Beispiel zu  
verwenden!

# Typen von Exceptions



```
try
{
    Console.WriteLine("Bitte geben Sie den Dividend ein: ");
    dividend = int.Parse(Console.ReadLine());
    Console.WriteLine("Bitte geben Sie den Divisor ein: ");
    divisor = int.Parse(Console.ReadLine());
    // Achtung: in dem Fall darf nicht auf double gecastet werden, da die Exception
    // sonst nicht auftritt!
    Console.WriteLine("Der Wert des Quotienten lautet: {0}", dividend / divisor);
    Console.ReadLine();
}
catch (DivideByZeroException)
{
    Console.WriteLine("Sie können durch 0 nicht dividieren!");
}
catch (FormatException)
{
    Console.WriteLine("Sie haben keine gültige Zahl eingegeben!");
}
```

2 Arten von Ausnahmen  
werden behandelt