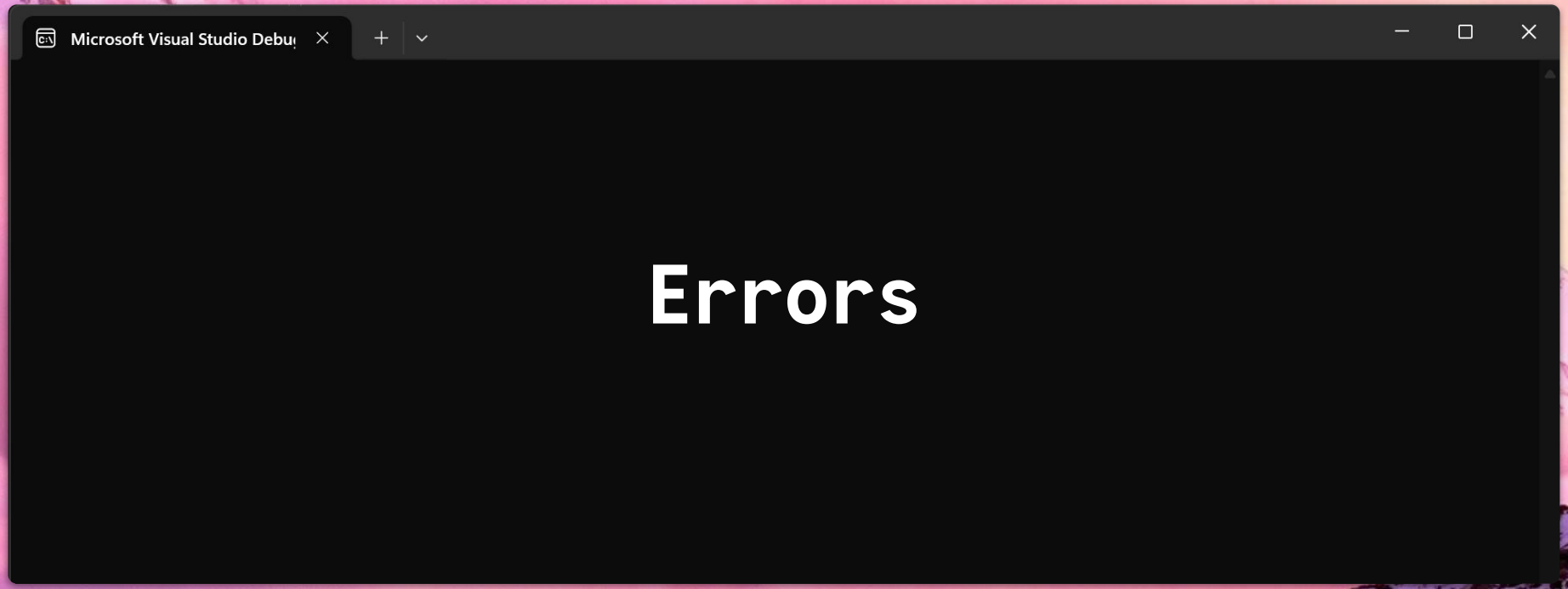

Microsoft Visual Studio Debu
Trying not to Crash Out...
Week 8
C:\Users\brend\source\repos\VGATestProject\x64\Release\VGATestProject.exe (process 70484) exited with code 0.
Press any key to close this window . . .


Microsoft Visual Studio Debu
Errors

Microsoft Visual Studio Debu

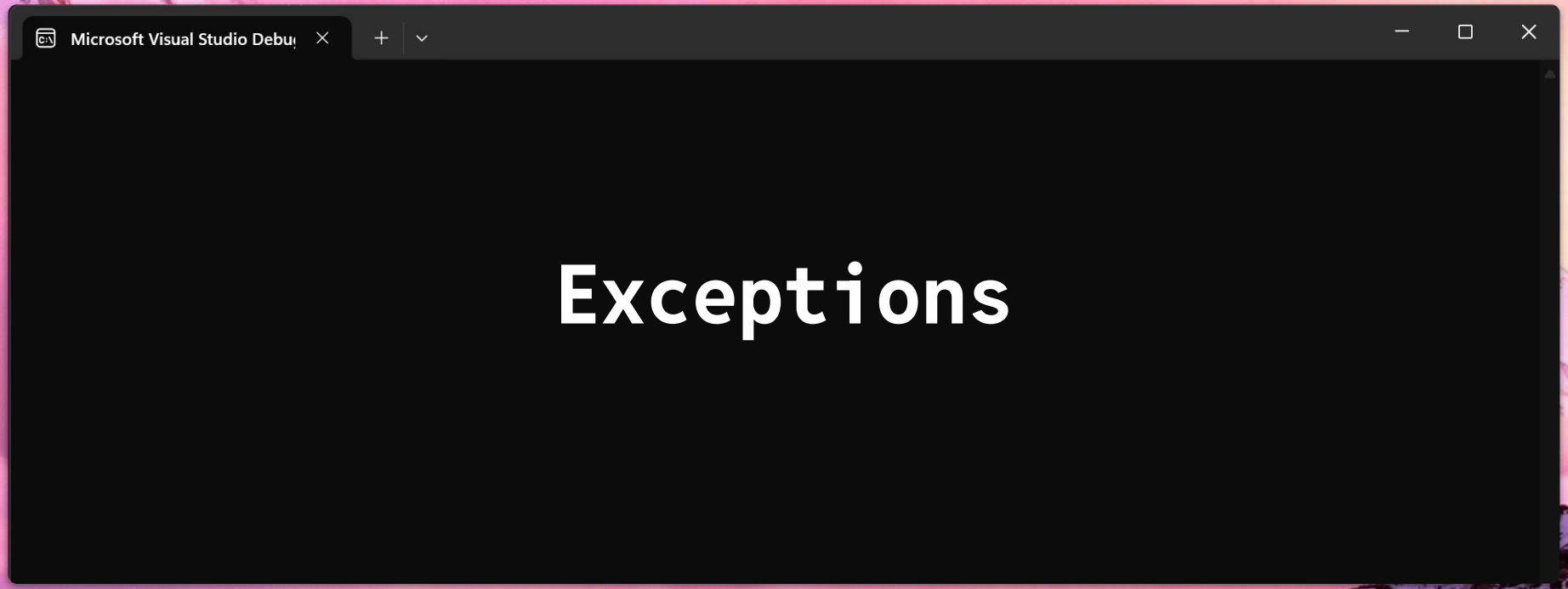
Compiler Misfire

If you get an *error*, that means something broke in your code when it was compiling. Code with errors can’t compile or run.

The main file may still run, but the code in the unbuilt file will be inaccessible

- IDEs like Visual Studio prevent the entire project from running if there is any error

```
C:\Users\brend\source\repos\VGATestProject\x64\Release\VGATestProject.exe (process 70484) exited with code 0.
Press any key to close this window . . .
```


Microsoft Visual Studio Debu
Exceptions

Microsoft Visual Studio Debu

“Houston, We Have a Problem”

If you get an *exception*, that means something broke in your code while it was running. (Sometimes called “Runtime Errors”)

Unlike Java, which automatically throws exceptions, C++ will still try to perform the operations. Depending on the compiler and specific operation you may see:

Program Crash, Strange Output Values (+inf, etc), etc

```
C:\Users\brend\source\repos\VGATestProject\x64\Release\VGATestProject.exe (process 70484) exited with code 0.
Press any key to close this window . . .
```

Microsoft Visual Studio Debu

Defining the Undefined

Exception handling controls the result of these behaviors:

```
#include <stdexcept>

throw std::runtime_error(“Cannot divide by 0!”);

try { ... /*Code contains throw command*/ }

catch (const std::runtime_error error) { ... }

// Throw can use any variable, including ints or strings instead
```

```
C:\Users\brend\source\repos\VGATestProject\x64\Release\VGATestProject.exe (process 70484) exited with code 0.
Press any key to close this window . . .
```

Microsoft Visual Studio Debu

<stdexcept> Types

logic_error -> exception class to indicate violations of logical preconditions or class invariants

invalid_argument -> exception class to report invalid arguments

domain_error -> exception class to report domain errors

length_error -> exception class to report attempts to exceed maximum allowed size

out_of_range -> exception class to report arguments outside of expected range

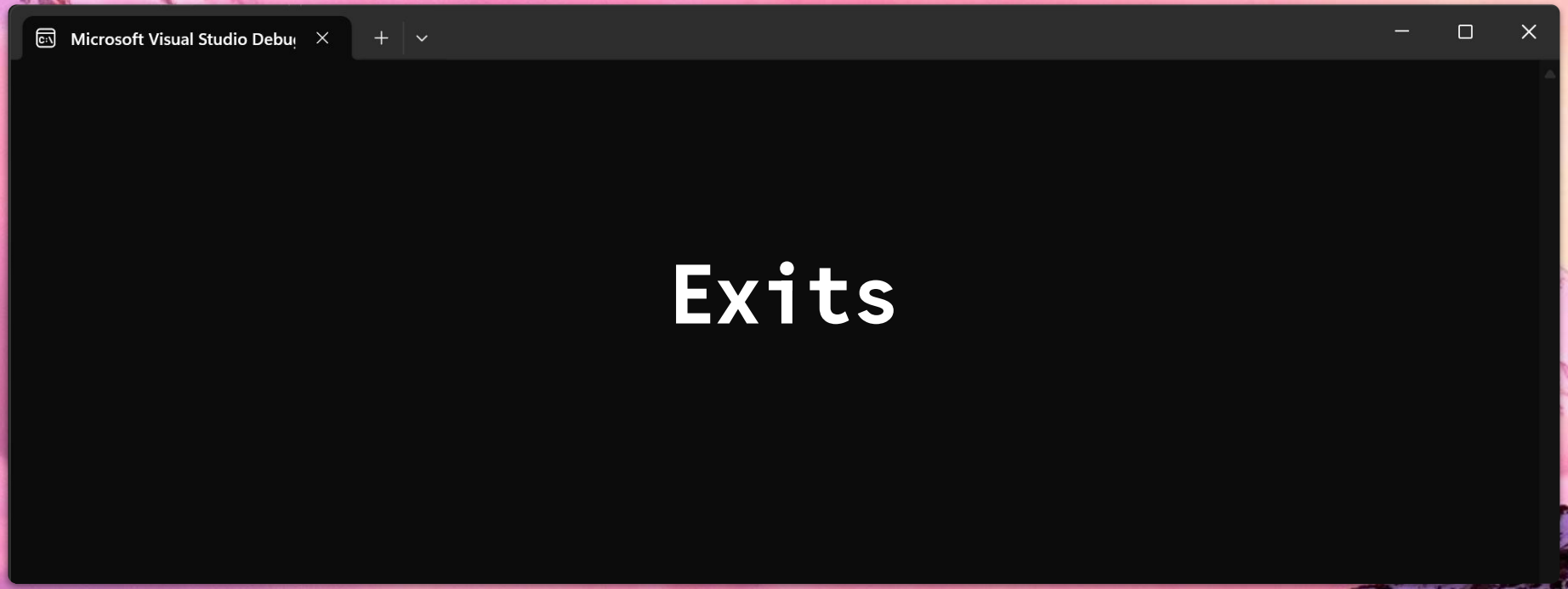
runtime_error -> exception class to indicate conditions only detectable at run time

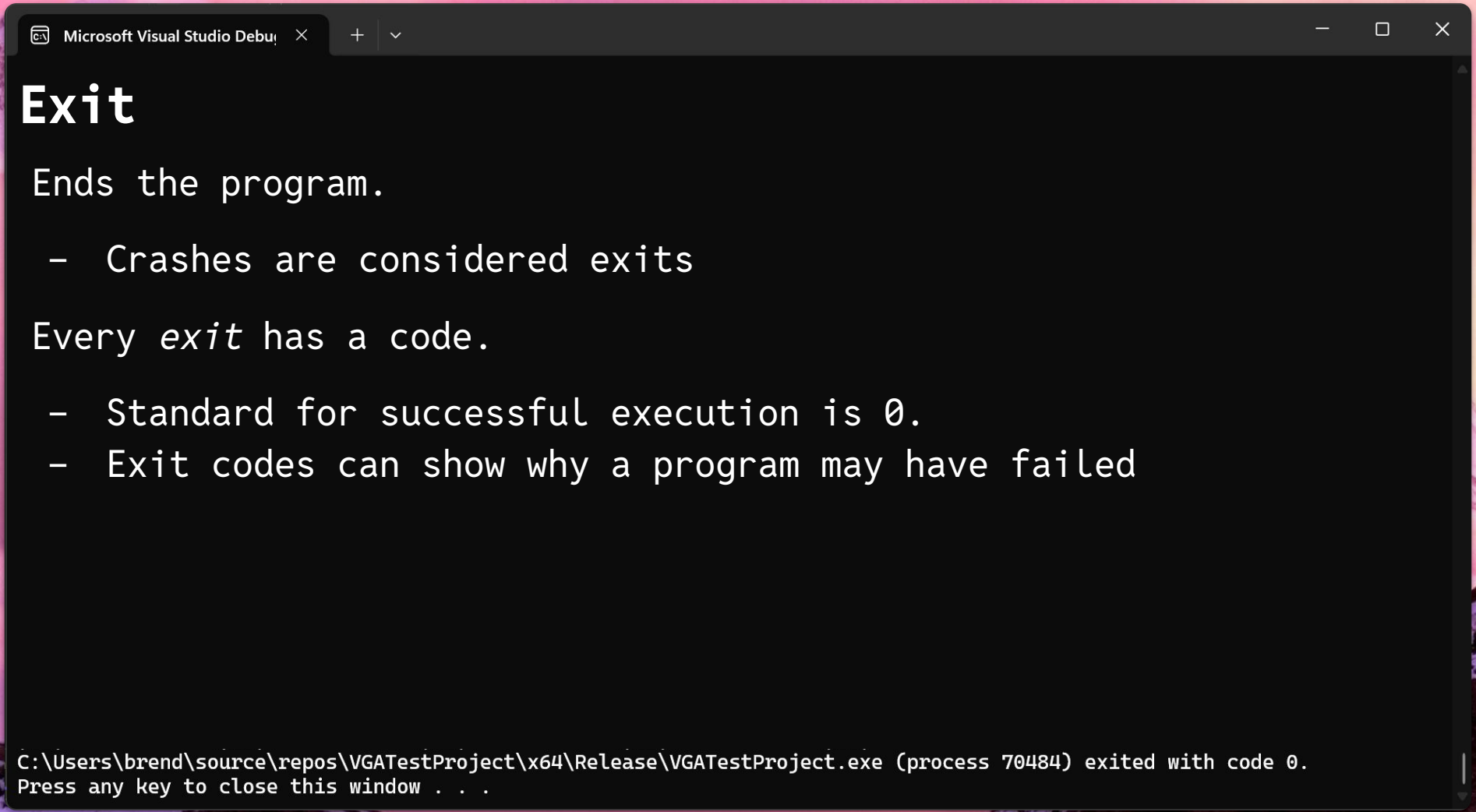
range_error -> exception class to report range errors in internal computations

overflow_error -> exception class to report arithmetic overflows

underflow_error -> exception class to report arithmetic underflows

```
C:\Users\brend\source\repos\VGATestProject\x64\Release\VGATestProject.exe (process 70484) exited with code 0.
Press any key to close this window . . .
```


Microsoft Visual Studio Debu
Exits


Microsoft Visual Studio Debu
Exit
Ends the program.
- Crashes are considered exits
Every *exit* has a code.
- Standard for successful execution is 0.
- Exit codes can show why a program may have failed
C:\Users\brend\source\repos\VGATestProject\x64\Release\VGATestProject.exe (process 70484) exited with code 0.
Press any key to close this window . . .

Microsoft Visual Studio Debu

Is there a Door?

There are a few ways to exit a program:

```
int main() { return 0; } // Exits normally with code 0


#import <stdio.h>

std::exit(0); // Exits normally with code 0

std::abort(404); // Abruptly stops all processes with code 404
```

```
C:\Users\brend\source\repos\VGATestProject\x64\Release\VGATestProject.exe (process 70484) exited with code 0.
Press any key to close this window . . .
```

Microsoft Visual Studio Debu

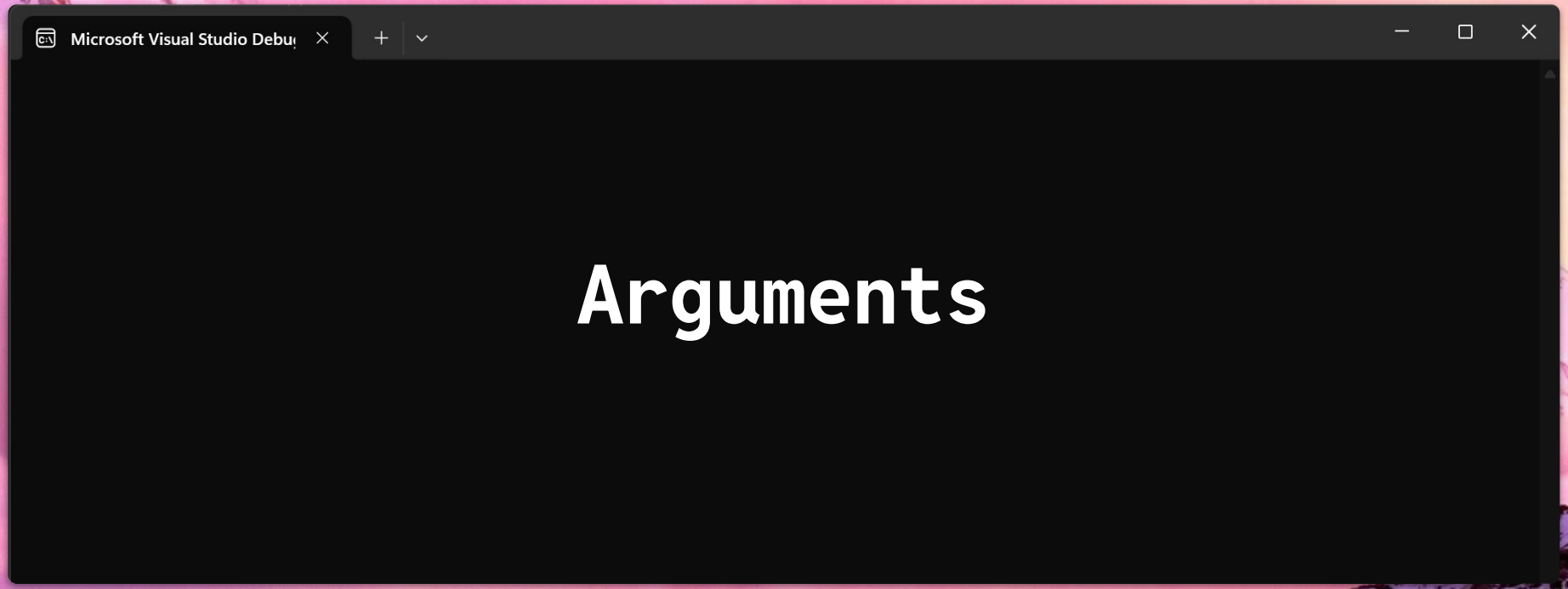
Modified Exit Behavior

Normally just deletes all data, then closes.

If you want to do extra stuff on program shutdown:

```
int atexit( function_to_call );
```

```
C:\Users\brend\source\repos\VGATestProject\x64\Release\VGATestProject.exe (process 70484) exited with code 0.
Press any key to close this window . . .
```


Microsoft Visual Studio Debu
Arguments

Microsoft Visual Studio Debu

Command Line Arguments

When you run a program, you can give it some extra info:

In terminal:

<path>/program.exe “String”

You can then access that string from program.exe

```
C:\Users\brend\source\repos\VGATestProject\x64\Release\VGATestProject.exe (process 70484) exited with code 0.
Press any key to close this window . . .
```

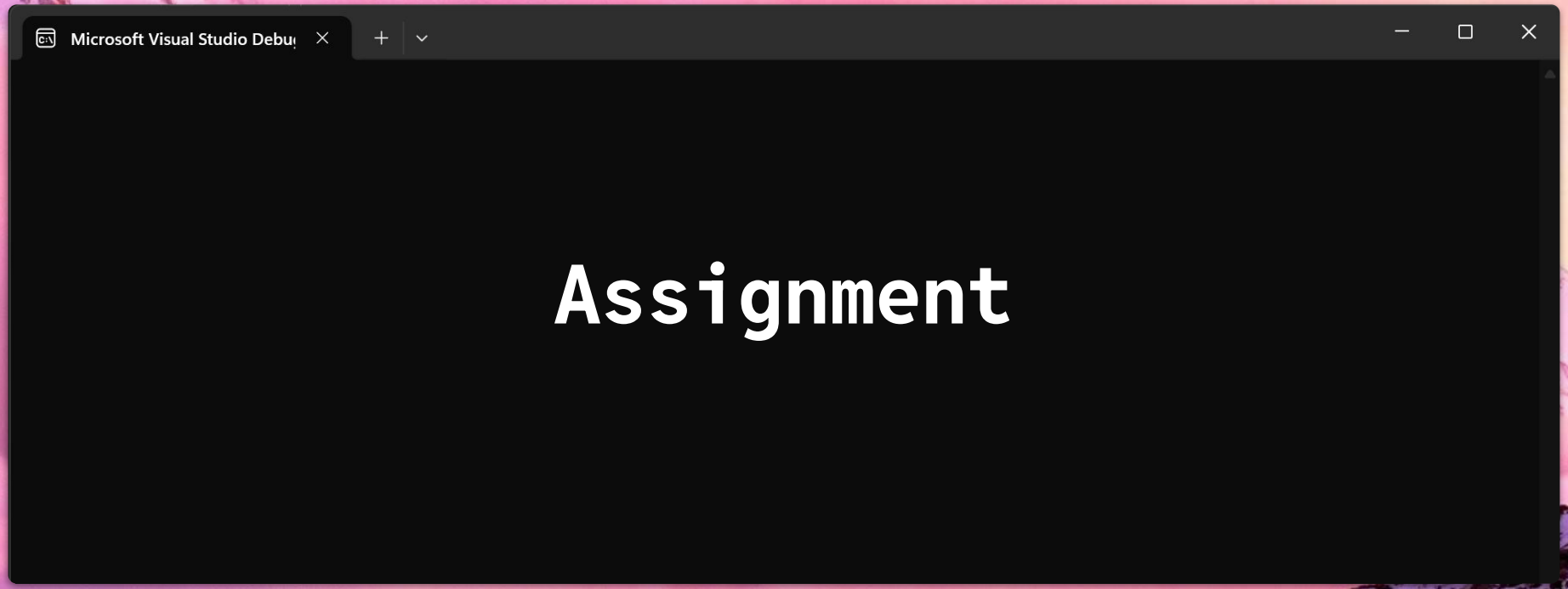
Microsoft Visual Studio Debu

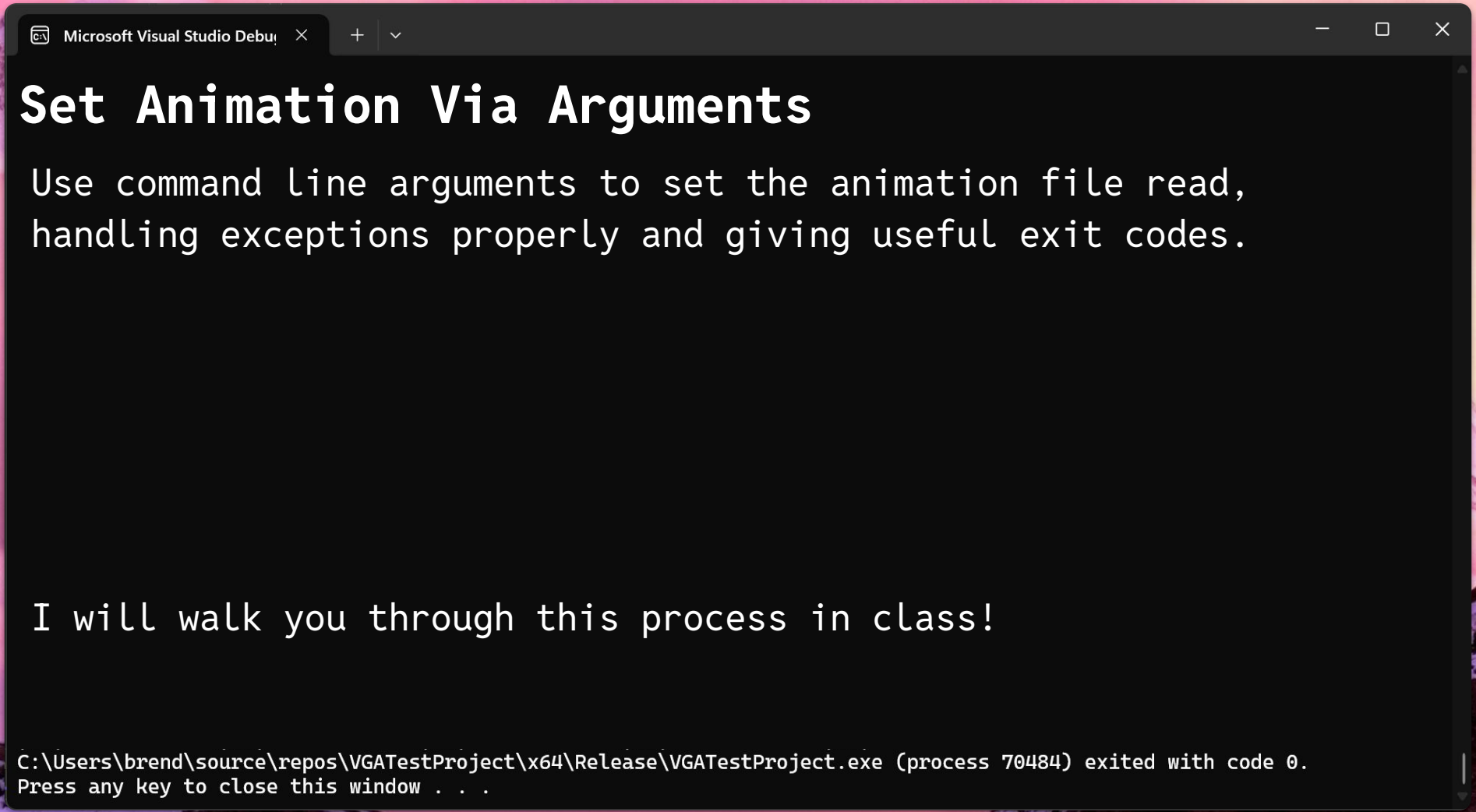
Example Code

```
int main (int argc, char* argv[]) {
    // argc is the number of arguments
    // argv is the actual argument data
    if (argc > 1) { filepath = argv[1]; }
}


// argv always has the program name at argv[0]
```

```
C:\Users\brend\source\repos\VGATestProject\x64\Release\VGATestProject.exe (process 70484) exited with code 0.
Press any key to close this window . . .
```


Microsoft Visual Studio Debu
Assignment


Microsoft Visual Studio Debu
Set Animation Via Arguments
Use command line arguments to set the animation file read,
handling exceptions properly and giving useful exit codes.
I will walk you through this process in class!
C:\Users\brend\source\repos\VGATestProject\x64\Release\VGATestProject.exe (process 70484) exited with code 0.
Press any key to close this window . . .