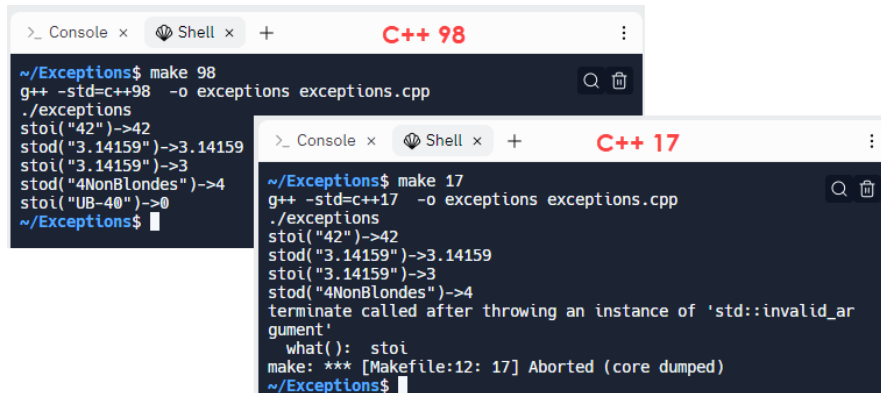


Handling Errors

We finished the last lesson with a question: what **should** the `stoi()` and `stod()` functions do when given invalid input? The C++ 17 version, from the standard library, does one thing, while the version we wrote does something else entirely when given the invalid input "UB-40".



The image shows two terminal windows side-by-side. The left window is titled 'C++ 98' and shows the output of a program compiled with g++ -std=c++98. It demonstrates that for the input 'UB-40', both `stoi` and `stod` return 0. The right window is titled 'C++ 17' and shows the output of a program compiled with g++ -std=c++17. For the same input 'UB-40', the program terminates with an exception: `terminate called after throwing an instance of 'std::invalid_argument'`. The `what()` function shows the error was thrown from `stoi`. The `make` command output indicates the program was aborted (core dumped).

```
~/Exceptions$ make 98
g++ -std=c++98 -o exceptions exceptions.cpp
./exceptions
stoi("42")->42
stod("3.14159")->3.14159
stoi("3.14159")->3
stod("4NonBlondes")->4
stoi("UB-40")->0
~/Exceptions$
```

```
~/Exceptions$ make 17
g++ -std=c++17 -o exceptions exceptions.cpp
./exceptions
stoi("42")->42
stod("3.14159")->3.14159
stoi("3.14159")->3
stod("4NonBlondes")->4
terminate called after throwing an instance of 'std::invalid_ar
gument'
what(): stoi
make: *** [Makefile:12: 17] Aborted (core dumped)
~/Exceptions$
```

To answer this question, first consider what **should** happen when you try to:

- Print the square root of -2?
- Open a file that doesn't exist? Read data from that file?
- Convert a string that doesn't contain a number to a number?

Each of these is **handled in a different** way. None of them are syntax or linker errors. Instead, they are **runtime errors**. The compiler and linker produced an executable, but when it runs, an error occurs.

Let's examine a few ways to **handle** such errors.



This course content is offered under a CC Attribution Non-Commercial license. Content in this course can be considered under this license unless otherwise noted.