# (8-1) Exception Handling in C++
# D & D Chapter 17

Instructor - Andrew S. O'Fallon
CptS 122 (July 22, 2024)
Washington State University

# Key Concepts

- Exceptions
- Exception handling
- `try` block
- `catch` handler
- Keyword `throw`
- Fault-tolerant programs



# What is an Exception?

- A *signal* that a problem has occurred during program execution that requires special processing
- Exceptions should only occur during “exceptional” circumstances
- C++ provides a mechanism for handling exceptions so that programs don’t just “crash” or stop executing, without a chance to recover



# What is Exception Handling?

- A process for detecting and resolving exceptions
- C++ exception handling is built on three keywords
  - `try`
  - `catch`
  - `throw`



# When to Use Exception Handling?

- Exception handling processes *synchronous* errors, which occur when a statement in the program executes
- Exception handling does *not* process *asynchronous* events that may happen independent of program flow



# `try` Block

- Contains code that might generate an exception





# `catch` Handler

- Executes as a result of an exception
- The correct handler is “activated” when a match occurs between the type of exception thrown and type of parameter for the handler
- An exception parameter should always be declared as a reference to the type of exception in the handler



# Keyword `throw`

- When an unexpected circumstance occurs an exception is generated by keyword `throw`





# Standard Library Exception Classes and Hierarchy (I)

- p. 775, Deitel & Deitel, C++ How To Program, 10th Ed.

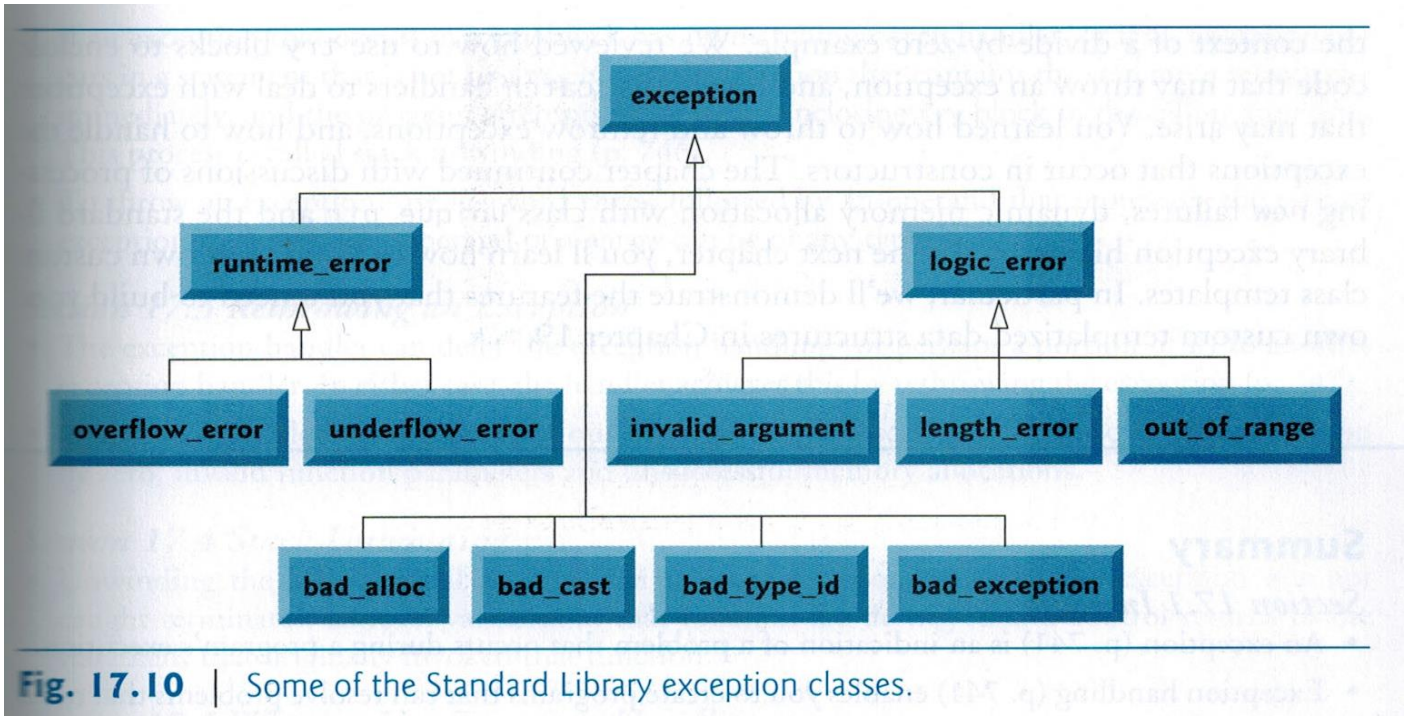


Fig. 17.10 | Some of the Standard Library exception classes.





# Standard Library Exception Classes and Hierarchy (II)

- We can write classes, which are derived from the standard library exception classes (Note: the standard exception classes are located in `<stdexcept>`)
  - `class` DivideByZeroException : `public` runtime_error



# Dividing-by-zero Example

- Provided in class





# Fault-tolerant Programs

- Programs that can satisfy most, if not all requirements, even if faults or exceptions occur
- These programs handle faults or exceptions gracefully, which provides a level of robustness



# Summary

- Exception handling provides a mechanism for building robust and fault-tolerant programs



# References


- P.J. Deitel & H.M. Deitel, *C++ How to Program (10th Ed.)*, Pearson Education , Inc., 2017.
- J.R. Hanly & E.B. Koffman, *Problem Solving and Program Design in C (7th Ed.)*, Addison-Wesley, 2013