

COMP3021 Java Programming

Exceptions vs. Assertions vs. if-else

- **Exceptions**

- Used to handle events that arise and disrupt the normal flow of the program during the execution of the program. (i.e. Something might happen)
- **Mainly deal with users' problems and unexpected system problems**
- Examples of usage:
 - Checking parameters passed to public / protected methods and constructors
 - Expecting the client code to recover from an exceptional situation
 - Addressing problems that might occur

- **Assertions**

- Used to check something that **MUST BE TRUE**. If it is not, then our program is broken and cannot recover from a broken program. We need to fix the bugs.
- **Mainly deal with programmers' problems (tackle programming bugs)**
- Examples of usage:
 - Check pre-conditions, i.e. check related conditions at the start of any method
 - Check post-conditions, i.e. check constraints at the end of any kind of method
 - Enforce constraints on arguments to private methods
 - State things that we know to be true, i.e. check for conditional cases that should never happen

- **if-else Statements**

- Used to control program flow

Answer of questions on page 50 of Exception Handling notes

- | | |
|--------------|--|
| a) if-else | (Since it refers to program flow control) |
| b) Exception | (Assume it is a public method, then anyone could call it with an invalid argument – users' problems) |
| c) Assertion | (Programming issue) |
| d) Exception | (System problem) |
| e) Assertion | (Programming issue, something that we know to be true for sure) |