Asserzioni

- Asserzione classica: assert(expr)
  - Richiede l'inclusione di cassert
  - Se "expr" non è true, il programma termina via abort()
  - è una macro, disabilitata se la costante NDEBUG è definita
- Asserzioni C++11: static_assert(expr, s)
  - "expr" è una constexpr booleana, deve essere valutabile a compile time
  - "s" è una stringa, descrizione dell'errore di compilazione risultante

Eccezioni

- Correttezza dell'accesso alle risorse e dell'esecuzione
  - Aiuta a gestire approcci alternativi (soluzioni fallback) nei casi eccezionali
- Segnala che il flusso di esecuzione non ha modo di proseguire
  - "throw" di una variabile di qualunque tipo
- Il "throw" di una eccezione causa lo "stack unwinding"
  - Si risale nello stack di esecuzione alla ricerca di codice che gestisca l'eccezione
- Un blocco "try" segnala al compilatore che il codice associato può tirare una eccezione
- Un blocco "catch" associato ad un try pemette di riprendere l'esecuzione interrotta dal throw
  - Per gestire tutte le possibili eccezioni si indica "…" nella clausola catch
- Si preferisce usare eccezioni di tipo std::exception (include exception) o derivate
  - throw by value, catch by reference
  - Il metodo virtuale what() ritorna il messaggio associato