

# Obrada izuzetaka

- ❖ Tretman izuzetaka, odnosno izuzetnih situacija koje su detektovane, može da bude na jednom od sledećih nivoa sigurnosti (*safety*), od najstrožijeg ka najlabavijem:
  - *Nothrow* (ili *nofail*) garancija: potprogram nikada ne baca izuzetak (funkcije označene kao *noexcept*); to znači da nema preduslove i uvek će sigurno očuvati sve invarijante i ispuniti svoje postuslove, ili pak greške sakriva ili tretira na drugi način; ovakvo ponašanje očekuje se od destruktora (oni su implicitno *noexcept*) i *move* konstruktora i operatora dodele, kao i drugih funkcija koje se implicitno pozivaju tokom prosleđivanja bačenog izuzetka do hvatača
  - *Jaka garancija sigurnosti od izuzetaka* (*strong exception safety guarantee*): ako potprogram baci izuzetak, stanje programa biće vraćeno na ono pre poziva tog potprograma (tzv. *rollback*) - sam potprogram je napravljen tako da to stanje povрати ili očuva
  - *Osnovna garancija sigurnosti od izuzetaka* (*basic exception safety guarantee*): ako potprogram baci izuzetak, stanje programa biće validno (iako može biti različito od onog pre poziva tog potprograma), tj. sve invarijante biće očuvane
  - *Nikakva garancija*: ukoliko se baci izuzetak, program može ostati u nekorektnom stanju; ovakve pojave predstavljaju *bagove* (*bug*), odnosno nepravilnosti u programu koje treba rešavati
- ❖ Preporuka je da svaka funkcija podrži najstrožiji od ovih redom iznesenih nivoa koji je moguće ispuniti

# Deo III: Detalji jezika C++

- ❖ Elementi i principi prevođenja i povezivanja
- ❖ Tipovi i konverzije
- ❖ Deklaracije i opseg važenja
- ❖ Izrazi
- ❖ Funkcije
- ❖ Životni vek varijabli
- ❖ Konstruktori, destruktori i operator dodele
- ❖ Objekti sa zauzetim resursima
- ❖ Inicijalizacija
- ❖ Preklapanje operatora