Simboličko izvršavanje programa

- cilj: automatsko pronalaženje kvarova generiranjem ispitnih slučajeva
- zadatak: potrebno proći kroz sve (što veći) broj puteva
- tek se odnedavno koristi zbog razvoja SAT rješavača i rješavača ograničenja
- problemi: grananje, pretlje, raspon vrijednosti varijabli, istovremenost, memorija,...
- značajke
    1. za ulazne vrijednosti koriste se simboličke umjesto konkretnih vrijednosti
    2. var. programa prikazuju se kao simbolički izrazi nad simboličkim vrijednostima
    3. kod ispitivanja simboličko izvršavanje koristi za gen. ul. pod. za ostvarive puteve
    - tijekom izvođenja dodaju se nova ograničenja nad varijablama
- u svakom trenutku održava:
    - o (sigma) - simboličko stanje - dotad posjećene var. -> simb. izrazi
    - SPC - simboličko ograničenje puta - posjećene grane, FOL bez kvant.
- obrada ograničenja
    - pokreće se **rješavač ograničenja** koji pronalazi pridruživanje varijablama
        - na mjestima grananja - da se izbjegnu istraživanja ako su ogr. nezadovoljiva
        - na mjestu pogreške - da se utvrde konkretne vrijednosti pogreške
    - koriste se SMT-rješavači
    - beskonačne petlje - potrebno postaviti ogr. na pretragu puteva

SMT-rješavači

- odlučuju o zadovoljivosti temeljne formule FOL u odnosu na pozadinske teorije
    - temeljna formula - ne sadrži slobodne varijable
- samostalni i brzi, uspješno rade s 100k varijabli i 100k linija koda
- teorija (logički dijelovi) - dio log. form. koji tvori ograničenje - pr. slajdovi 15.-18.
    - teorija jednakosti, peano aritmetika
- primjeri zadataka:
    - klasični problem raspoređivanja poslova - n poslova, svaki od m zadataka koji moraju završiti kad su počeli + max. vrijeme trajanja
        - ograničenja: postoji redoslijed zadataka u 1 poslu, dva zadatka koji trebaju isti stroj ne mogu biti pokrenuta istovremeno
- aritmetika razlike - podvrsta lin. aritmetike u kojoj su dozvoljene samo form. **t - s <= c** gdje su **t** i **s** varijable, a **c** numerička konstanta
    - može ih se svesti na aritm. razlike t.d. **z = 0, tj. s <= -c**
    - iz ograničenja u tom obliku oblikuje se **težinski usmjereni graf**
        - svaka var. je čvor, a t -s <= c je usmjereni brid od s do t uz težinu c
        - traže se negativni ciklusi u grafu za pretragu zadovoljivosti
    - pokazivanje zadovoljivosti - lazy offline pristup
        - ograničenja se pretvore u SAT problem
        - ako je SAT nezadovoljiv => problem nezad., inače SMT provjerava SAT model

- kombiniranje teorija - vrlo težak problem, i odlučivost i konzistentnost, ne samo rješenje

Konkretno/simboličko izvršavanje

- problem simboličkog izvršavanja - ispitni slučaj ne može se generirati ako SMT-rješavač ne može učinkovito razriješiti ograničenje - nemamo garanciju
  -> rješenje: kombinacija konkretnog i simboličkog izvršavanja
- pristupi kombiniranja:
  - **konkoličko ispitivanje** - konkretno upravlja simboličkim, izvodi se konkr. i pamte simboličke vrijednosti
  - **izvršavanjem generirano ispitivanje** - prije svake oper. provjerava se jesu li sve vrijednosti konkretne - ako jesu -> konkretno izvršavanje, inače -> simboličko
- primjeri - 31.-35.

Heuristike, optimizacije, alati

- problem eksplozije broja puteva, a nije ih puno ostvarivo
- heuristike pretraživanja puta
  - slučajni put ako su oba ostvariva na grananju
  - statički graf kontrolnog puta - istraživanje najbližeg dotad nepokrivenog
- problm rješavanja ograničenja - SMT treba što manje pozivati
  - uklanjanje nebitnih ograničenja
- istovremeno izvođenje
  - cilj otkriti redundantna izvršavanja i utrke za resursima
- alati za simb. verif.: DART, CUTE, EXE i KLEE, S2E, JPF
- SMT-rješavači: Z3, Yices 2, CVC4, MathSAT5, Boolector