Otkrivanje redudantnih test primera

Seminarski rad u okviru kursa
Verifikacija softvera
Matematički fakultet

Una Stanković, Mirko Brkušanin, Miloš Samardžija
una_stankovic@yahoo.com, mi13211@alas.matf.bg.ac.rs, miloss208@gmail.com

8. maj 2018.

Sažetak

U ovom tekstu je ukratko prikazan proces kreiranja metodologije za otkrivanje redudantnih test primera. Autori su u njemu izneli proces razmišljanja i testiranja sa ciljem dolaska do određenih zaključaka i potencijalnog rešenja problema.

Sadržaj

1 Uvod

Redudantni test primeri su oni test primeri koji pokrivaju iste delove koda ili pokrivaju delimično iste delove koda, tj. njihovo pokrivanje se preklapa.

Pronalaženje redudantnih test primera predstavlja veoma bitan deo testiranja softvera kojem do danas nije pridodavan veliki značaj. Razlog koji stoji iza toga je što još uvek nije razvijena adekvatna metodologija koja bi programerima i testerima omogućila da bez dodatnog troška (posebno vremenskog) otkriju test primere koji su redudantni, a potom ih i uklone.

Značaj otkrivanja redudantnih test primera se posebno ističe u projektima otvorenog koda u kojima svi učesnici u stvaranju istog (a može ih biti i nekoliko stotina) prilikom dodavanja novih delova koda dodaju i nove test primere, bez ikakve provere da li takvi test primeri već postoje i da li se tim test primerima pokrivaju neki delovi koda koji su već pokriveni drugim test primerima. Iz tog razloga broj testova za određeni kod može narasti do nerazumnih granica, što još više otežava rad nad softverom otvorenog koda, kao i otkrivanje grešaka, propusta i slično.

U radu će biti iznet detaljan proces razmišljanja, testiranja i kreiranja metodologije za otkrivanje ovakvih test primera, sa posebnim osvrtom na alat llc u okviru LLVM-a.

2 Eksperimenti

U ovoj sekciji će biti izneti osnovni eksperimenti koje smo vršili kako bismo došli do određenih zaključaka.

2.1 Jednostavni primeri

Najpre, pokušaćemo sa dodavanjem nekoliko jednostavnih primera, koji su, pre svega, jednostavni za analizu, a potom ćemo posmatrati ponašanje primenom različitih test primera, od kojih će neki biti redudantni.

2.1.1 Maksimum 4 broja

Prvi primer koji posmatramo je otkrivanje maksimuma 4 broja. Kod je napisan u programskom jeziku C.

```
1  //This is just a simple example on which we should try to
       think about all the test examples and observe the
       results for them
2  #include <stdio.h>
3
4  int main( void )
5  {
6      int a, b, c, d;
7      int largest, smallest;
8
9      printf( "Enter four integers (separate them with
       spaces): " );
10     scanf( "%d %d %d %d", &a, &b, &c, &d );
```

```
11
12      largest = smallest = a;
13
14      if ( largest < b ){
15          largest = b;
16      }
17      else if ( b < smallest ){
18          smallest = b;
19      }
20      if ( largest < c ){
21          largest = c;
22      }
23      else if ( c < smallest ){
24          smallest = c;
25      }
26      if ( largest < d ){
27          largest = d;
28      }
29      else if ( d < smallest ){
30          smallest = d;
31      }
32
33      printf( "\nLargest: %d\n", largest );
34      printf( "Smallest: %d", smallest );
35
36      return 0;
37  }
```

Sada, kreirajmo neke test primere:

```
4
3
2
1
```

3 Zaključak