

Univerzitet u Sarajevu
Elektrotehnički fakultet u Sarajevu
Odsjek za računarstvo i informatiku

Zadaća 5
Računarske arhitekture
Dženana Huseinspahić, Esma Karahodža
i Dženeta Kudumović

26.04.2020.

Sadržaj

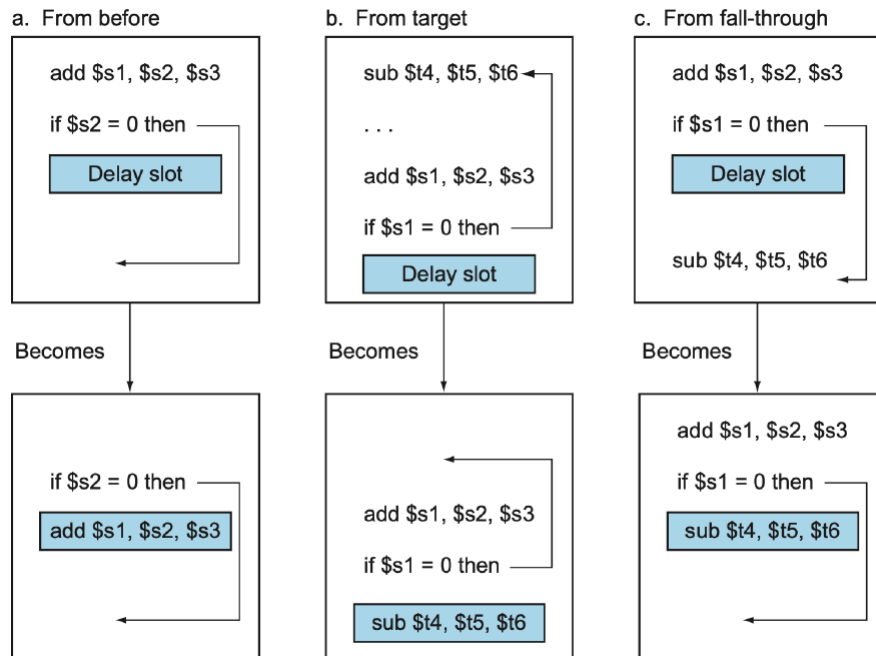
1	Uvod	2
2	Opis projekta	3
3	Upute za korištenje	5
4	Testiranje	6
4.1	Test 1	6
4.2	Test 2	6
4.3	Test 3	6
4.4	Test 4	7
5	Raspored zaduženja	8

1 Uvod

Pri kompajliranju programa u fazi prevoenja u MIPS instrukcije na jednociklusnim procesorima dolazi do zastoja na mjestima gdje se nalaze instrukcije grananja (bne, bqe, itd.). Naime, dok se odlučuje kojim će se putem nastaviti izvršavanje procesor ne može nastaviti sa radom. Ovo odlučivanje potraje nekoliko ciklusa što zbog prirode jednociklusnog procesora znatno produžava ukupno vrijeme izvršavanja programa. Cilj instrukcijâ zadrške je da procesor zaposle za vrijeme izvršavanja instrukcije grananja. Ovakav način izvršavanja programa se zove odgoeno grananje.

Bitno je napomenuti i da instrukcija zadrške ne može biti bilo koja instrukcija. Neophodno je da se ona može uvijek izvršiti da ne bi došlo do greške u rezultatima.

Instrukcija odgode može biti prikazana na tri načina prikazana na sljedećoj slici.



Razlika se ogleda u tome da li se instrukcija zadrške bira od instrukcija koje se nalaze iznad, ispod ili unutar instrukcije grananja. Načini odabira su ovdje nazvani A, B i C, respektivno.

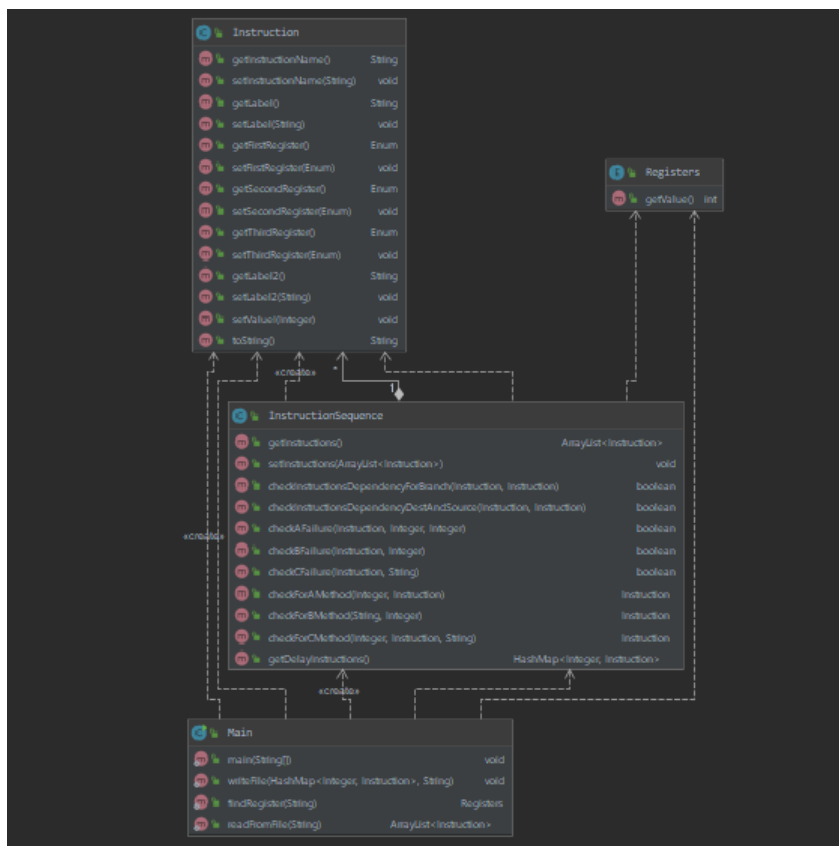
2 Opis projekta

Cilj ovog projekta (zadaje) je da za datu sekvencu MIPS instrukcija pronađe instrukcije zadržke i da ih zapiše u novu datoteku zajedno sa rednim brojem instrukcije grananja za koju je zadržka.

Program se sastoji od 3 klase i jednog enuma:

- Main
- Registers
- Instruction
- InstructionSequence

U nastavku se može vidjeti klasni dijagram.



U enumu **Registers** se nalazi pobrojanih svih 31 registara zajedno sa metodomama koje omogućavaju pristup registru po njegovom rednom broju.

Bazna klasa **Instruction** se sastoji od raznih atributa i konstruktora koji pokrivaju svaki slučaj i formu koju može imati jedna MIPS instrukcija. Također

se prema JavaBeans specifikaciji u ovoj klasi nalaze i seteri i geteri. Poslije toga još imamo i `toString` metodu koja omogućava formatiran ispis svake instrukcije.

Klasa **InstructionSequence** kao atribut sadrži `ArrayList` instrukcija koji se na početku postavlja na sve pročitane instrukcije. Glavna metoda ove klase je `getDelayInstructions` koja pronalazi instrukcije zadrške koristeći pomoćne metode i vraća hash mapu kojoj je ključ redni broj instrukcije kojoj je bila potrebna zadržka a vrijednost je instrukcija zadrške.

Za metod A traženja instrukcije zadrške metoda pretražuje sve instrukcije iznad (pri čemu se prednost daje instrukciji koja je bliža instrukciji grananja) i traži instrukciju koja je nezavisna (ta nezavisnost se ogleda u tome što se odredišni registar te instrukcije ne koristi u instrukciji grananja). Program takoer ne uzima ni instrukcije koje imaju za odredišni registar registar koji se u instrukcijama izmeu pronaene i instrukcije grananja koristi kao izvorišni bez obzira što takva instrukcija ispunjava ostale uvjete. U ovoj metodi kao instrukcija zadrške se takoer ne uzimaju ni instrukcije grananja jer to dodatno komplikuje izvršavanje.

Za metod B je uzeta instrukcija koja se nalazi na adresi koja predstavlja odredište grananja. Takodjer, obezbjedjen je i slučaj kada grananje ode u neočekivanom smjeru tj. kada do grananja ne dodje. To je obezbjeeno tako što kao instrukcija zadrške ne može biti uzeta instrukcija koja kao odredišni registar koristi registar koji je izvorišni u nekoj od instrukcija u nizu instrukcija koje se nalaze iza početne instrukcije grananja.

Za metod C je uzeta jedna od instrukcija koje slijede iza instrukcije grananja pri čemu se prednost daje bližoj instrukciji. Takodjer, obezbjedjen je slučaj kada grananje ode u neočkivanom smjeru tj. kada dodje do grananja, jer se ne uzimaju instrukcije čiji se odredišni registar pojavljuje kao izvorišni u nizu instrukcija koje slijede iza adrese koja predstavlja odredište grananja.

Klasa **Main** se pored `main` metode sastoji od metoda `readFromFile` i `writeFile` koje obavljaju istoimene akcije. Metoda `writeFile` prima hash mapu koju vraća klasa `InstructionSequence` a `readFromFile` vraća niz instrukcija koje su spremne za čitanje.

Napomena 1: Za sve metode traženja instrukcije zadrške postoji mogućnost da dodje do hazarda druge vrste (a ne kontrolnog hazarda) jer nisu provjeravane sve ostale instrukcije, tj. moguće da se pomjeranjem neke od instrukcija zadrške poremeti tašnost tog dijela skevence instrukcija.

Napomena 2: Pri upisu u datoteku instrukcija `sw` i `lw` umjesno brojčane vrijednosti offseta je napisana samo riječ da bi se dobilo na jednostavnosti pri učitavanju instrukcija.

Napomena 3: Instrukcije `j`, `jal` i `jr` neće nikada biti uzete kao zadržka.

3 Upute za korištenje

Datoteka koja sadrži set instrukcija iz kojih će se tražiti zadrška mora biti formatirana na određeni način, inače program neće raditi kako je očekivano. Uslovi za pisanje datoteke su sljedeći:

- Instrukcije moraju biti napisane jedna ispod druge
- Komentari u MIPS kodu nisu dozvoljeni
- Instrukcije moraju biti napisane u formatu: naziv instrukcije, razmak, registar, zarez, razmak, registar, zarez, razmak itd. Instrukcije `sw` i `lw` se pišu sa `offsetom` i `zagrdom` kao i uobičajeno
- Nazivi instrukcija nisu case sensitive
- Registri se primaju po rednom broju `$r1-$r31`
- Registri se primaju u četiri različita oblika (`$R3`, `R3`, `$r3`, `r3`)
- Labele za instrukcije su dozvoljene, s tim da moraju počinjati tekстом, inače će biti pročitane kao registar
- Radi jednostavnosti, instrukcije grananja mogu biti samo `bqe` i `bne`

Naredna sekvenca instrukcija je primjer jednog od načina na koji datoteka sa ulaznim instrukcijama treba biti formatirana.

```
sub $r0, $r4, $r3
label1: beq R6, $r0, label2
or $r4, $r0, $r0
add $r0, $r5, $r3
beq $r6, r0, label1
label2: lw $r3, 0($r5)
```

Nakon pravilnog formatiranja ulazne datoteke, može se pokrenuti program. Program na ulazu traži da se u konzoli unese putanja na kojoj se nalazi datoteka sa instrukcijama. Potrebno je unijeti tačnu putanju (bez razmaka).

Program pri završetku dobijene instrukcije zadrške upisuje u novu datoteku sa imenom `zadrška.txt` na istoj lokaciji kao i ulazna datoteka. Ukoliko u datoj sekvenci instrukcije nije bilo moguće naći zadršku nova datoteka se neće kreirati i biće ispisana poruka na konzoli.

4 Testiranje

U nastavku je testirano nekoliko mogućih scenarija za pronalazak zadrške u MIPS sekvencama instrukcija.

4.1 Test 1

Sadržaj ulazne datoteke:

```
or r11, r4, r5
sub r4, r3, r6
bne r6, r4, label
addi r6, r20, 5
label: sub r6, r4, r7
```

Sadržaj izlazne datoteke:

Za 3.instrukciju predložena instr. zadrške: or \$r11, \$r4, r5

4.2 Test 2

Sadržaj ulazne datoteke:

```
add r2, r2, r3
label1: beq r9, r2, label2
or r2, r9, r9
label2: add r0, r4, r3
```

Sadržaj izlazne datoteke:

Za 2. instr. predložena instr. zadrške: label2: add \$r0, \$r4, r3

4.3 Test 3

Sadržaj ulazne datoteke:

```
lw $r9, 0($r0)
label1: beq $r1, $r9, label2
and $r2, $r5, $r7
add $r0, $r7, $r5
beq $r1, $r0, label1
label2: lw $r3, 0($r8)
```

Sadržaj izlazne datoteke:

Za 2.instr. predložena instr. zadrške: label2: lw \$r3, offset(\$r8)

Za 5.instr. predložena instr. zadrške: and \$r2, \$r5, \$r7

4.4 Test 4

Sadržaj ulazne datoteke:

```
lw $r3, 6($r4)
addi $r9, $r10, 23
sw $r5, 7($r6)
```

Izlazna datoteka u ovom slučaju nije ni kreirana a ispis na konzolu je sljedeći:
”Zadržka nije potrebna, jer data sekvenca instrukcija nema nijednu instrukciju grananja”.

5 Raspored zaduženja

Osmišljavanje projekta, njegovih klasa te metoda i općenito rješavanje bug-ova u kodu je raeno timski. Svaki član tima je imao lokalnu kopiju projekta koju je testirao. O izmjenama u kodu se redovno javljalo radi što brže izrade.

1) Dženana Huseinspahić

- dio main-a koji čita iz datoteke i formira date instrukcije
- prepravka metode writeFile tako da omogućuje ispis u datoteku u istom direktoriju kao i ulazna datoteka

2) Esma Karahodža

- bazne klase Instruction i Registers
- prva verzija metode writeFile koja upisuje rezultate u unaprijed zadatu datoteku

3) Dženeta Kudumović

- metode u klasi InstructionSequence koje pronalaze instrukcije zadržke
- dio main-a koji koristi napisane metode iz InstructionSequence