

Задатак за RA_155_2019

У оквиру пројектног задатка потребно је реализовати МАН преводац који преводи програме са вишег асемблерског језика на основни MIPS 32bit асемблерски језик. Преводац треба да подржава детекцију лексичких, синтаксних и семантичких грешака као и генерисање одговарајућих извештаја о евентуалним грешкама. Излаз из преводиоца треба да садржи коректан асемблерски код који је могуће извршавати на MIPS 32bit архитектури (симулатору).

Уз пројекат је неопходно испоручити и пропратну документацију која садржи:

- Анализу проблема
- Концепт решења
- Програмско решење
- Верификацију

МАН - Преводац

МАН (Мипс Асемблер Високог Нивоа) је алат који преводи програм написан на вишем MIPS 32bit асемблерском језику на основни асемблерски језик. Виши MIPS 32bit асемблерски језик служи лакшем асемблерском програмирању јер уводи концепт **регистарске** променљиве. Регистарске променљиве омогућавају програмерима да приликом писања инструкција користе променљиве уместо правих ресурса. Ово знатно олакшава програмирање јер програмер не мора да води рачуна о коришћеним регистрима и њиховом садржају. Ради једноставности пројекта у инструкцијама не смеју да се користе прави ресурсе већ искључиво променљиве.

МАН језик треба да подржава следеће MIPS инструкције:

1. la - (load address) учитавање адресе у регистар
2. bltz - (branch on less than zero) скок ако је регистар мањи од нуле
3. jalr - (jump and link register) скок на адресу из регистра и складиштење повратне адресе у \$ra
4. lw - (load word) читавање једне меморијске речи
5. sll - (shift left logical) логички померај улево
6. jr - (jump register) скок на адресу из регистра (return преко \$ra)
7. or - (or) битско или
8. add - (addition) сабирање
9. abs - (Absolute value) апсолутна вредност
10. b - (unconditional branch) безусловни скок

У наставку је дат пример граматике за синтаксу МАН језика:

$Q \rightarrow S ; L$	$S \rightarrow \text{_mem } mid \text{ } num$	$L \rightarrow eof$	$E \rightarrow \text{add } rid, rid, rid$
	$S \rightarrow \text{_reg } rid$	$L \rightarrow Q$	$E \rightarrow \text{addi } rid, rid, num$
	$S \rightarrow \text{_func } id$		$E \rightarrow \text{sub } rid, rid, rid$
	$S \rightarrow id: E$		$E \rightarrow \text{la } rid, mid$
	$S \rightarrow E$		$E \rightarrow \text{lw } rid, num(rid)$
			$E \rightarrow \text{li } rid, num$
			$E \rightarrow \text{sw } rid, num(rid)$
			$E \rightarrow \text{b } id$
			$E \rightarrow \text{bltz } rid, id$
			$E \rightarrow \text{nop}$

Терминални симболи МАН језика су:

: ; , ()

`_mem, _reg, _func num id, rid, mid, eof,`

као и имена свих инструкција, што би за горњи пример били:

`add, addi, sub, la, lw, li, sw, b, bltz, nor`

Овде се напомиње на који начин се врши декларисање функција, меморијских и регистарских променљивих:

- Декларација функције:

`_func funcName`

Пример:

```
1 _func foo
```

- Декларација меморијске променљиве:

`_mem varName value`

`varName` - мора почети малим словом **m** у наставку може бити било који број.

Пример:

```
1 _mem m4 55
2 _mem m78 12
```

- Декларација регистарске променљиве:

`_reg varName`

`varName` - мора почети малим словом **r** у наставку може бити било који број.

Пример:

```
1 _reg r1
2 _reg r9
```

За реализацију пројекта могу се користити приложене датотеке (директоријум *src*) за лексичку анализу као и предлог реализације променљивих и инструкција (*IR.h*). У прилогу се налази и пример улазног програма *simple.mavn* (директоријум *examples*).

У наставку је дат пример полазног програма и њему одговарајућег асемблерског излаза: Програм на МАН програмском језику:

```
1 _mem m1 6;
2 _mem m2 5;
3 _reg r1;
4 _reg r2;
5 _reg r3;
6 _reg r4;
7 _reg r5;
8 _func main;
9     la      r4,m1;
10    lw      r1, 0(r4);
11    la      r5, m2;
12    lw      r2, 0(r5);
13    add     r3, r1, r2;
```

Програм преведен на MIPS 32bit асемблерски језик:

```
1 .globl main
2
3 .data
4 m1:      .word 6
5 m2:      .word 5
6
7 .text
8 main:
9     la      $t0, m1
10    lw      $t1, 0($t0)
11    la      $t0, m2
12    lw      $t2, 0($t0)
13    add     $t0, $t1, $t2
```

Задаци

Задатак МАНН преводиоца је да уčitану улазну датотеку писану на МАНН језику преведе на MIPS 32bit асемблер. Ограничити се на једну улазну датотеку. Користити екстензију “.mavn” за улазну датотеку која садржи програм на МАНН језику.

Преводилац треба да приликом преводјења:

1. Додели ресурсе за регистарске променљиве - ограничити се на 4 регистра: `t0`, `t1`, `t2` и `t3` из MIPS архитектуре
2. Све меморијске променљиве генерише у секцију за податке - `.data` водећи рачуна о синтакси асемблерског језика
3. Све инструкције смести у програмску секцију - `.text`
4. Генерише излазну датотеку са екстензијом `.s` која садржи преведен и коректан MIPS 32bit асемблерски језик полазног програма

НАПОМЕНЕ:

1. Приликом преводјења омогућити детекцију и реаговање на лексичке, синтаксне и семантичке грешке.
2. Приложена граматика представља само пример граматике. Документација пројектног задатка мора садржати дефинисану граматiku у складу са Вашим задатим скупом инструкција.
3. Поред један приложен пример написати минимално још 3 тестна програма, од којих ће макар један изазвати синтаксну грешку, а један преливање у фази доделе ресурса.
4. Уколико се полазни програм успешно преведе на MIPS 32bit асемблерски језик, извршавање програма је могуће проверити коришћењем QtSpim симулатора који је доступан на линку.