

Санкт-Петербургский политехнический университет Петра Великого
Кафедра компьютерных систем и программных технологий

Отчёт по лабораторной работе

Дисциплина: Низкоуровневое программирование

Тема: RISC-V

Выполнил студент гр. 3530901/10005
Калашников О. Ю.

(подпись)

Преподаватель
Коренев Д.А.

(подпись)

“ ____ ” _____ 2022 г.

Санкт-Петербург
2022

Оглавление

ТЗ	3
Метод решения	3
Реализация программы	4
Результат работы программы	4
Реализация подпрограммы и тестовой программы	5
Тестовая программа	5
Подпрограмма	6
Результат работы программы	6
Вывод	6

ТЗ

Написать программу для машины RISC-V, которая реализует нахождение максимального элемента в массиве чисел. Необходимо также выделить основной функционал в подпрограмму и написать для неё тестовую программу.

Метод решения

Для нахождения максимального элемента массива необходимо циклически пройти каждый элемент, сравнивая его с максимальным значением, найденным на данный момент (для первого элемента текущее максимальное значение принимается за 0).

Например, рассмотрим массив [3, 5, 9, 4, 1]:

1. $3 > 0 \Rightarrow 3$ — текущий максимальный элемент;
2. $5 > 3 \Rightarrow 5$ — текущий максимальный элемент;
3. $9 > 5 \Rightarrow 9$ — текущий максимальный элемент;
4. $4 < 9 \Rightarrow 9$ — остаётся текущим максимальным элементом;
5. $1 < 9 \Rightarrow 9$ — остаётся текущим максимальным элементом;

Ответ: 9.

Реализация программы

```
1 .text
2 __start:
3 .globl __start
4
5     la a3, array_length
6     lw a3, 0(a3)           # загружаем длину массива
7
8     li a5, 0               # переменная под ответ
9
10    la a6, array           # загружаем адрес массива
11
12 loop:
13     addi a3, a3, -1        # отнимаем 1 от длины массива
14     lw t2, 0(a6)          # загружаем в t2 текущий элемент
15     bltu a5, t2, writeMax  # if t2 > a5 goto writeMax
16     beqz a3, exit         # если длина массива 0, то останов
17
18     addi a6, a6, 4         # записываем в a6 следующее значение массива
19     jal zero, loop        # начинаем новую итерацию массива
20
21 writeMax:
22     lw a5, 0(a6)          # записываем в a5 текущее максимальное значение массива
23     addi a6, a6, 4         # записываем в a6 следующее значение массива
24     jal zero, loop        # начинаем новую итерацию массива
25
26 exit:
27     li a0, 10             # записываем 10 в a0
28     ecall                # останов
29
30 .rodata
31 array_length:
32     .word 8
33
34 .rodata
35 array:
36     .word 6, 12, 18, 24, 48, 69, 70, 69
```

Результат работы программы

Возьмём массив [6, 12, 18, 24, 48, 69, 70, 69]. Ожидаемый результат — 70.

Mnemonic	Number	Value
a1	x11	0
a2	x12	0
a3	x13	0
a4	x14	0
a5	x15	70

Результат совпал с ожидаемым.

Реализация подпрограммы и тестовой программы

Тестовая программа

```
1 # тестовая программа
2 .text
3 __start:
4 .globl __start
5
6     la a3, array_length
7     lw a3, 0(a3)           # загружаем длину массива
8
9     li a5, 0               # переменная под ответ
10
11    la a6, array            # загружаем адрес массива
12
13    call find_max           # вызов подпрограммы
14
15    li a0, 10               # записываем 10 в a0
16    ecall                  # останов
17
18 .rodata
19 array_length:
20     .word 8
21
22 .rodata
23 array:
24     .word 6, 12, 18, 24, 48, 69, 70, 69
```

Подпрограмма

```
1 # подпрограмма
2 .text
3 find_max:
4 .globl find_max
5
6 loop:
7     addi a3, a3, -1      # отнимаем 1 от длины массива
8     lw t2, 0(a6)        # загружаем в t2 текущий элемент
9     bltu a5, t2, writeMax # if t2 > a6 goto writeMax
10    beqz a3, exit        # если длина массива 0, то останов
11
12    addi a6, a6, 4        # записываем в a6 следующее значение массива
13    jal zero, loop       # начинаем новую итерацию массива
14
15 writeMax:
16     lw a5, 0(a6)        # записываем в a5 текущее максимальное значение массива
17     addi a6, a6, 4      # записываем в a6 следующее значение массива
18     jal zero, loop     # начинаем новую итерацию массива
19
20 exit:
21     ret
```

Результат работы программы

Возьмём массив [6, 12, 18, 24, 48, 69, 70, 69]. Ожидаемый результат — 70.

Mnemonic	Number	Value
a1	x11	0
a2	x12	0
a3	x13	0
a4	x14	0
a5	x15	70

Результат совпал с ожидаемым.

Вывод

В ходе выполнения лабораторной работы мной была разработана программы для машины RISC-V, в том числе и для реализации с подпрограммой, которые реализуют нахождение максимального элемента в заданном массиве чисел. Результаты работы программы для обоих загрузчиков совпали с ожидаемыми.