

Санкт-Петербургский политехнический университет Петра Великого
Высшая школа интеллектуальных систем и суперкомпьютерных технологий

Отчёт по лабораторной работе № 3

Дисциплина: Низкоуровневое программирование

Тема: Программирование RISC-V

Выполнил студент гр.3530901/10005 Севдат Туфаногуллари

Преподаватель

“ ” _____

Санкт-Петербург

2022

Оглавление

1. Техническое задание
2. Метод решения
3. Руководство программисту
4. Реализация программы 1
5. Работа программы 1
6. Реализация программы 2 с подпрограммой
7. Работа программы 2

1. Техническое задание

Реализовать нахождение наибольшего общего делителя (НОД) для массива чисел при помощи EDSAC.

2. Метод решения

Идея алгоритма заключается в том, что мы вычитаем из большего числа меньшее и заменяем первое на их разность до тех пор, пока их разность не станет равна нулю. В таком случае уменьшаемое и вычитаемое как раз и будут искомым числом.

Способ решения

Наименьший число, найденное в списке, вычитается из каждого числа до тех пор, пока не будет найдено значение, которое при вычитании из каждого числа в списке приведет к нулю. Пример:

- $A = 10; B = 2$
 - $A - B = 8$
 - $A - B = 6$
 - $A - B = 4$
 - $A - B = 2$
 - $A - B = 0$
- Ответ = 2
-

3. Руководство программисту

Программа должна иметь размер списка и сам список, чтобы найти наибольшее число, которое можно разделить для каждого элемента списка. Размер списка сохраняется в переменной в t2, а адрес первого элемента сохраняется в t3. Ответ сохраняется в t4. Переменные t2 и t3 являются константами и не изменяются на протяжении всей программы.

4. Реализация программы 1

```
painP.s ×
1 .globl __start
2 .rodata
3 array_length:
4 .word 5
5 .data
6 array:
7 .word 10, 5, 20, 60, 200
8 .text
9
10 __start:
11
12 la t0, array #firstElementAddress
13 la t1, array_length #sizeAddress
14 lw t1, 0(t1) #size
15 li a2, 0 #count
16 li a3, 0 #save
17
18 smallest:
19 bgeu a2, t1, finish
20 lw a4, 0(t0) #firstElement 10
21 beq a2, zero, addSmallest
22 bgeu a3, a4, addSmallest
23 addi t0, t0, 4
24 addi a2, a2, 1
25 j smallest
26 addSmallest:
27 mv a3, a4
28 addi t0, t0, 4
29 addi a2, a2, 1
30 j smallest
```

```

30 j smallest
31 finish:
32
33 mv a5,a3 #smallestInList
34 addi a5,a5,1
35
36 restart:
37 addi a5,a5,-1
38 la t0, array #firstElementAdress
39 la t1, array_length #sizeAdress
40 lw t1, 0(t1) #size
41 li a2, 0 #count
42 li a3, 0 #save
43
44 nextNumber:
45 lw a4, 0(t0) #firstElement 10
46 beq a2,t1,end
47
48 addition:
49 add a3,a3,a5
50 beq a4,a3, possible
51 bgeu a3,a4, restart
52 j addition
53
54 possible:
55 li a3, 0 #save
56 addi t0,t0,4
57 addi a2,a2,1
58 j nextNumber
59

```

```

58 j nextNumber
59
60
61 end:
62 mv a1,a5
63 li a0,1
64 ecall
65 li a0, 10
66 ecall

```

5. Работа программы 1

Программа выполнена при таких входных данных:

```
2 .rodata
3 array_length:
4 .word 5
5 .data
6 array:
7 .word 10, 5, 20, 60, 200
8 .text
```

Jupyter

File Edit Run Settings Help

Editor Simulator

▶ ⏮ ⏪ ⏩ ⏭ 📄 🧠

Bkpt	Address	Machine Code	Basic Code	Source Code	Registers	Memory	Cache
					Mnemonic	Number	Value
<input type="checkbox"/>	0x00010090	0x00000693	addi x13, x0, 0	li a3, 0	a1	x11	0x00000005
<input type="checkbox"/>	0x00010094	0x00428293	addi x5, x5, 4	addi t0, t0, 4	a2	x12	0x00000005
<input type="checkbox"/>	0x00010098	0x00160613	addi x12, x12, 1	addi a2, a2, 1	a3	x13	0x00000000
<input type="checkbox"/>	0x0001009c	0xfddff06f	jal x0, -36	j nextNumber	a4	x14	0x00000000
<input type="checkbox"/>	0x000100a0	0x00078593	addi x11, x15, 0	mv a1, a5	a5	x15	0x00000005
<input type="checkbox"/>	0x000100a4	0x00100513	addi x10, x0, 1	li a0, 1	a6	x16	0x00000000
<input type="checkbox"/>	0x000100a8	0x00000073	ecall	ecall	a7	x17	0x00000000
<input type="checkbox"/>	0x000100ac	0x00a00513	addi x10, x0, 10	li a0, 10	s2	x18	0x00000000
<input type="checkbox"/>	0x000100b0	0x00000073	ecall	ecall			

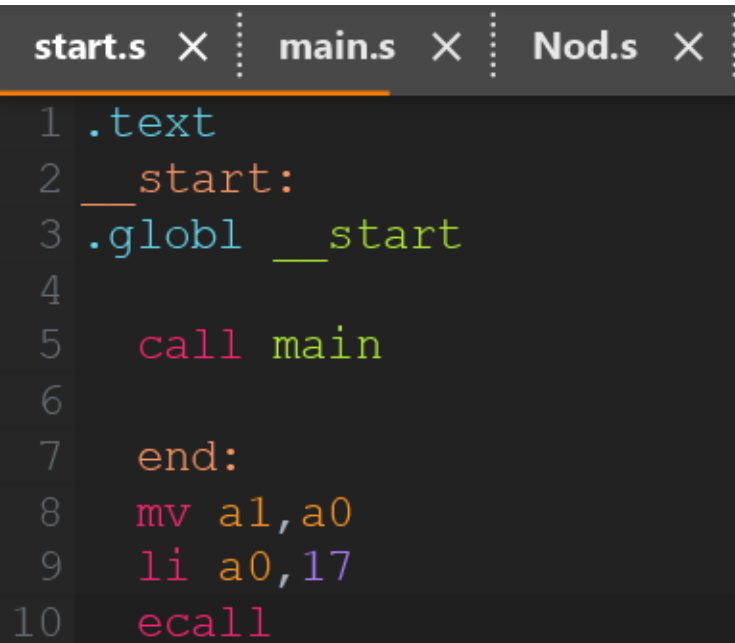
Integer (X) Floating (F)

Console

5

Ответ: a5 x15 0x00000005

6. Реализация программы 2 с подпрограммой



The image shows a screenshot of an IDE with three tabs: start.s, main.s, and Nod.s. The start.s tab is active and displays assembly code. The code is as follows:

```
1 .text
2 __start:
3 .globl __start
4
5     call main
6
7     end:
8     mv a1,a0
9     li a0,17
10    ecall
```

start.s X | main.s X | Nod.s X |

```
1 .rodata
2 array_length:
3 .word 5
4 .data
5 array:
6 .word 10, 5, 20, 60, 200
7 .text
8
9 main:
10 .globl main
11
12 la a0, array #firstElementAdress
13 la a1, array_length #sizeAdress
14 lw a1, 0(a1) #size
15
16 addi sp, sp, -16 #выделение памяти в стеке
17 sw ra, 12(sp) # сохранение
18 call Nod
19 lw ra, 12(sp) # восстанавливаем
20 addi sp, sp, 16 # освобождение памяти в сетке
21
22 end:
23 mv a1, a5
24 li a0, 1
25 ecall
26 ret
```


start.s × | main.s × | **Nod.s** ×

```
1 .text
2 Nod:
3 .globl Nod
4 mv t0,a0
5 mv t1,a1
6
7 li a2, 0      #count
8 li a3, 0      #save
9
10 smallest:
11 bgeu a2, t1, finish
12 lw a4, 0(t0) #firstElement 10
13 beq a2, zero, addSmallest
14 bgeu a3,a4, addSmallest
15 addi t0, t0, 4
16 addi a2, a2, 1
17 j smallest
18 addSmallest:
19 mv a3, a4
20 addi t0, t0, 4
21 addi a2, a2, 1
22 j smallest
23 finish:
24
25 mv a5,a3 #smallestInList
26 addi a5,a5,1
27
28 restart:
29 addi a5,a5,-1
30 mv t0,a0
```

```
30 mv t0,a0
31 mv t1,a1
32 li a2, 0      #count
33 li a3, 0      #save
34
35 nextNumber:
36 lw a4, 0(t0) #firstElement 10
37 beq a2,t1,end
38
39 addition:
40 add a3,a3,a5
41 beq a4,a3, possible
42 bgeu a3,a4, restart
43 j addition
44
45 possible:
46 li a3, 0      #save
47 addi t0,t0,4
48 addi a2,a2,1
49 j nextNumber
50 end:
51 ret
```

7. Работа программы 2

Программа выполнена при таких входных данных:

```
2 .rodata
3 array_length:
4 .word 5
5 .data
6 array:
7 .word 10, 5, 20, 60, 200
8 .text
```

Jupyter

File Edit Run Settings Help

Editor Simulator

Registers Memory Cache

Bkpt	Address	Machine Code	Basic Code	Source Code	Mnemonic	Number	Value
<input type="checkbox"/>	0x00010000	0x00000317	auipc x6, 0	auipc x6, 0	zero	x0	0x00000000
<input type="checkbox"/>	0x00010004	0x00830067	jalr x0, x6, 8	jalr x0, x6, 8	ra	x1	0x00010010
<input type="checkbox"/>	0x00010008	0x00000317	auipc x6, 0	call main	sp	x2	0xbffffff0
<input type="checkbox"/>	0x0001000c	0x014300e7	jalr x1, x6, 20	call main	gp	x3	0x10008000
<input type="checkbox"/>	0x00010010	0x00050593	addi x11, x10, 0	mv a1, a0	tp	x4	0x00000000
<input type="checkbox"/>	0x00010014	0x01100513	addi x10, x0, 17	li a0, 17	t0	x5	0x000100f4
<input type="checkbox"/>	0x00010018	0x00000073	ecall	ecall	t1	x6	0x00000005
<input type="checkbox"/>	0x0001001c	0x00000517	auipc x10, 0	la a0, array	t2	x7	0x00000000
<input type="checkbox"/>	0x00010020	0x0c450513	addi x10, x10, 196	la a0, array			

Integer (X) Floating (F)

Console

5

Ответ: a5 x15 0x00000005