

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

Отчёт по лабораторной работе № 3

Дисциплина: Низкоуровневое программирование

Тема: Программирование RISC-V

Вариант: 9

Выполнил студент гр. 3530901/00002 _____ М.А. Разин
(подпись)

Принял преподаватель _____ Д.С. Степанов
(подпись)

“ ____ ” _____ 2021 г.

Санкт-Петербург
2021

Задача:

1) Разработать программу на языке ассемблера RISC-V, реализующую определенную вариантом задания функциональность, отладить программу в симуляторе VSim/Jupiter. Массив (массивы) данных и другие параметры (преобразуемое число, длина массива, параметр статистики и пр.) располагаются в памяти по фиксированным адресам

2) Выделить определенную вариантом задания функциональность в подпрограмму, организованную в соответствии с ABI, разработать использующую ее тестовую программу. Адрес обрабатываемого массива данных и другие значения передавать через параметры подпрограммы в соответствии с ABI. Тестовая программа должна состоять из инициализирующего кода, кода завершения, подпрограммы main и тестируемой подпрограммы.

Вариант задания:

Вариант: 9 - Расчет биномиальных коэффициентов для данного показателя по треугольнику Паскаля.

Алгоритм:

```
fun BinCoeff(n: Int): List<Int> {  
    var a = 1  
    val res = mutableListOf<Int>()  
    for (t in 1..n + 1) {  
        res += a  
        a = a * (n + 1 - t) / t  
    }  
    return res  
}
```

Рис. 1 Расчет n-ой строки Треугольника Паскаля, Kotlin.

Выполнение работы:

Показатель располагается под меткой “power”

Массив – под меткой “array”

```

bincoeff.s X
1 .text
2 start:
3 .globl start
4     la a6, array      #a6 <адрес 0 эл массива>
5     addi a6, a6, 4     #a6 = a6 + 4 <адрес 1 эл>
6     lw a5, power      # a5 = n <показатель>
7     li a4, 1          # a4 = a <текущий коэфф>
8     li a3, 1          # a3 = t <счётчик цикла>
9 loop:
10    addi t0, a5, 1
11    sub t0, t0, a3
12    mul t0, t0, a4
13    div a4, t0, a3
14    sw a4, 0(a6)
15    addi a6, a6, 4
16    addi a3, a3, 1
17    bgt a3, a5, finish
18    jal zero, loop
19 finish:
20    li a0, 10
21    ecall
22 .rodata
23 power:
24 .word 5
25 .data
26 array:
27 .word 1

```

Рис. 2 Код программы для задачи 1

0x00010068	0	0	0	1
0x00010064	0	0	0	5
0x00010060	0	0	0	10
0x0001005c	0	0	0	10
0x00010058	0	0	0	5
0x00010054	0	0	0	1

Рис. 3 Результат выполнения программы

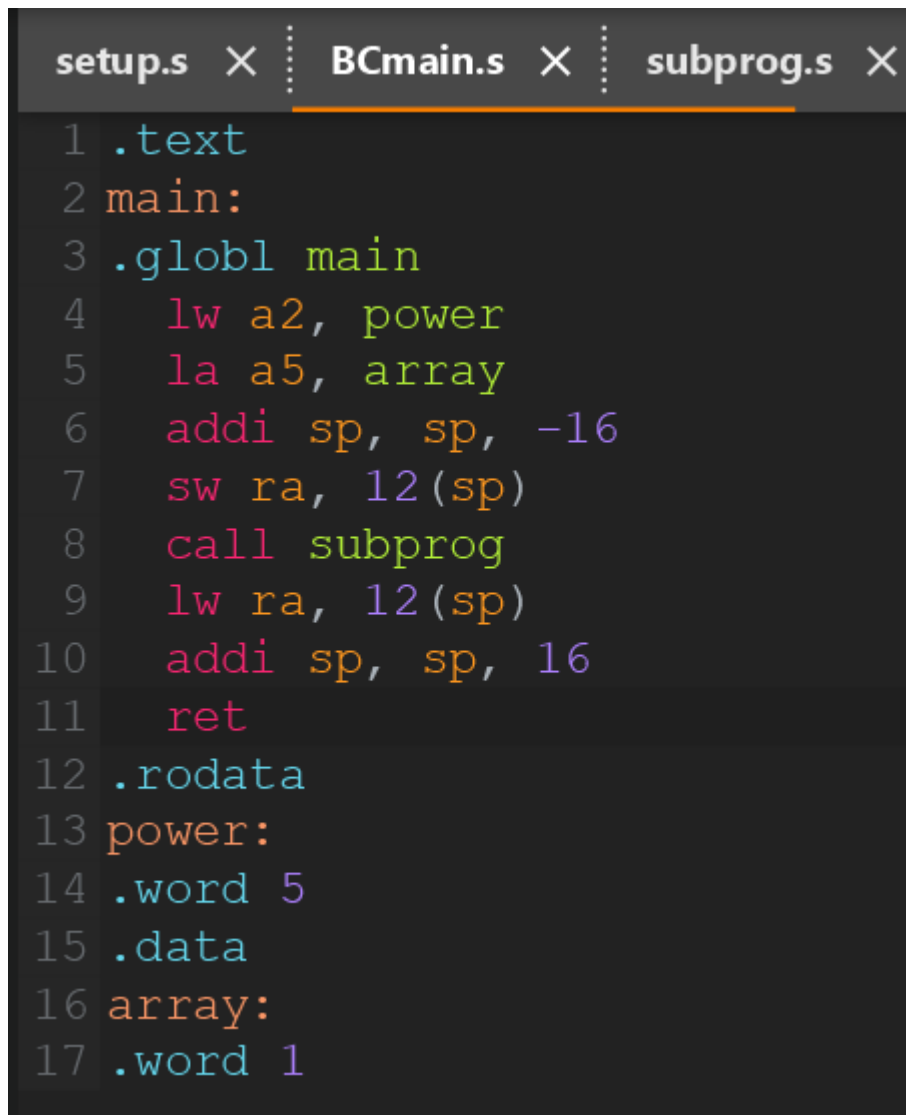
Аналогичный алгоритм используется для задачи 2 и в целом часть подпрограмма базируется на изменённой версии кода для задачи 1. Так же разработана вызывающая ее тестовую программа.

```

1 .text
2 start:
3 .globl start
4 call main
5 exit:
6 li a0, 10
7 ecall

```

Рис. 4 Код тестовой программы для задачи 2



```
1 .text
2 main:
3 .globl main
4     lw a2, power
5     la a5, array
6     addi sp, sp, -16
7     sw ra, 12(sp)
8     call subprog
9     lw ra, 12(sp)
10    addi sp, sp, 16
11    ret
12 .rodata
13 power:
14 .word 5
15 .data
16 array:
17 .word 1
```

Рис. 5 Код программы main для задачи 2

```

BCmain.s × subprog.s ×
1 .text
2 subprog:
3 .globl subprog
4 # a2 = <показатель>
5 # a5 = <адрес 0 эл>
6     addi a6, a5, 4 #a6 = a5 + 4 <адрес 1 эл>
7     li a4, 1      # a4 = a <текущий коэфф>
8     li a3, 1      # a3 = t <счётчик цикла>
9 loop:
10    addi t0, a2, 1
11    sub t0, t0, a3
12    mul t0, t0, a4
13    div a4, t0, a3
14    sw a4, 0(a6)
15    addi a6, a6, 4
16    addi a3, a3, 1
17    bgt a3, a2, finish
18    jal zero, loop
19 finish:
20    ret

```

Рис. 6 Код подпрограммы для задачи 2

0x00010090	0	0	0	1
0x0001008c	0	0	0	5
0x00010088	0	0	0	10
0x00010084	0	0	0	10
0x00010080	0	0	0	5
0x0001007c	0	0	0	1

Рис. 7 Результат выполнения программы

Вывод:

В ходе выполнения лабораторной работы была разработана программа на языке ассемблера RISC-V, выполняющая расчет биномиальных коэффициентов для данного показателя по треугольнику Паскаля и выполняющая запуск как цельной программы, так и подпрограммы, организованную в соответствии с ABI.

Список использованных источников:

http://kspt.icc.spbstu.ru/media/files/2020/lowlevelprog/riscv_prgc.pdf

http://kspt.icc.spbstu.ru/media/files/2020/lowlevelprog/riscv_subprgc.pdf

<https://github.com/riscv-non-isa/riscv-asm-manual/blob/master/riscv-asm.md>