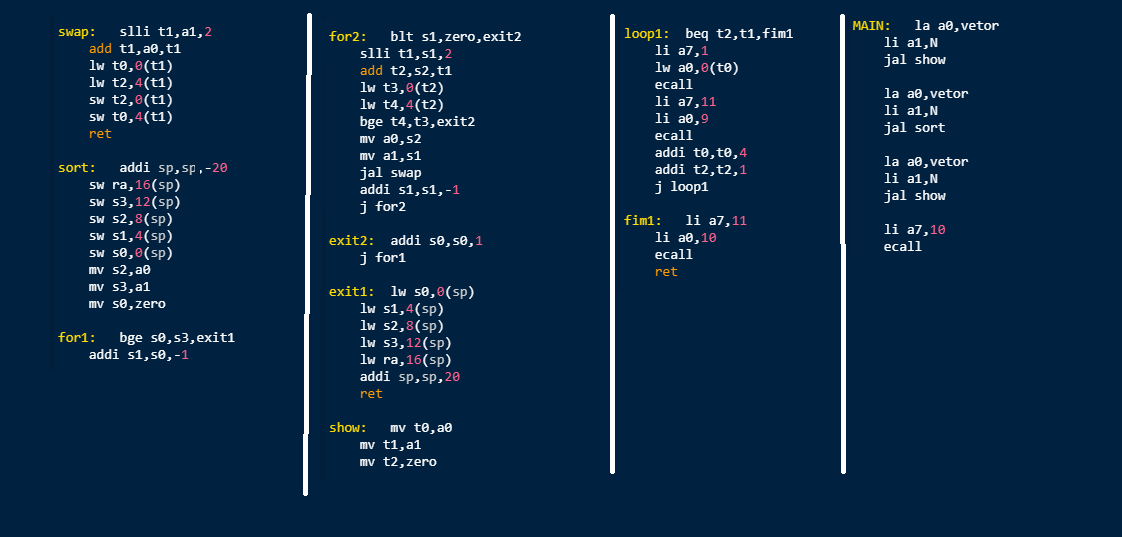
|  |  |
| --- | --- |
| unb | Universidade de Brasília  Organização e Arquitetura de Computadores  Alunos : Rafael Oliveira de Souza - 15/0081537 |

**Laboratório 1**

*Assembly RISC-V*

**Questão 1**

**Sort.s**



(1.1)

Qual o tamanho em bytes do código executável? 943 bytes

Qual o tamanho memória de dados usada?

O RARS possui um contador de instruções que resultou nos dados apresentados abaixo:

Vetor = [9,2,5,1,8,2,4,3,6,7,10,2,32,54,2,12,6,3,1,78,54,23,1,54,2,65,3,6,55,31]

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Type | J | R | I | S | B | U | Total |
| Instruções | 405 | 999 | 1554 | 317 | 462 | 3 | 3740 |

(1.2)

-----------------------------------------------------------------------------------------------------------------------------

a) n = [9,2,5,1,8,2,4,3,6,7,10,2,32,54,2,12,6,3,1,78,54,23,1,54,2,65,3,6,55,31]

t(n) = 3740 \* 50 \* 10-6

t(n) = 0,187

n0 = [1,1,1,2,2,2,2,2,3,3,3,4,5,6,6,6,7,8,9,10,12,23,31,32,54,54,54,55,65,78]

t0(n) = 942 \* 50 \* 10-6

t0(n) = 0,0471

n1 = [78,65,55,54,54,54,32,31,23,12,10,9,8,7,6,6,6,5,4,3,3,3,2,2,2,2,2,1,1,1]

t1(n) = 8291 \* 50 \* 10-6

t1(n) = 0,41455

---------------------------------------------------------------------------------------------------------------------------

b)

CPI = 1

CLOCK = 50

Vetor = [10,20,30,40,50,60,70,80,90,100]

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Type | J | R | I | S | B | U | Total |
| Instruções |  |  |  |  |  |  | 4537 |

Tempo de execução = Nos de ciclos \* Freq. processador

Tempo de execução = 4537 \* 50 \* 10-6

Tempo de execução = 0,22685

Vetor = [100,90,80,70,60,50,40,30,20,10]

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Type | J | R | I | S | B | U | Total |
| Instruções | 583 | 1517 | 2072 | 495 | 632 | 3 | 5302 |

Tempo de execução = Nos de ciclos \* Freq. processador

Tempo de execução = 5302 \* 50 \* 10-6

Tempo de execução = 0,2651

**Questão 2**

*Diretivas de Otimização*

O0 (default) - Reduz o tempo de compilação e gera mensagens de debug.

O1 - Busca diminuir o tamanho do código e o tempo de execução.

O2 - Melhora o tempo de compilação e a performance do código gerado quando comparada com a diretiva O1.

***Flags Ativadas***

|  |  |  |
| --- | --- | --- |
| O1 | O2 | O3 |
| -fipa-pure-const | Todas de O0 | Todas de O1 |
| - fipa-profile | -fthread-jumps | -finline-functions |
| -fipa-reference | -falign-functions | -funswitch-loops |
| -fmerge-constants | -falign-jumps | -fpredictive-commoning |
| -fmove-loop-invariants | -falign-loops | -fgcse-after-reload |
| -freorder-blocks | -falign-labels | -ftree-loop-vectorize |
| -fshrink-wrap | -fcaller-saves | -ftree-loop-distribute-patterns |
| -fsplit-wide-types | -fcrossjumping | -ftree-slp-vectorize |
| -ftree-bit-ccp | -fcse-follow-jumps | -fvect-costmodel |
| -ftree-ccp | -fcse-follow-jumps | -ftree-partial-pre |
| -fssa-phiopt | -fdelete-null-pointer-checks | -fipa-cp-clone options |
| -ftree-ch | -fexpensive-optimizations |  |
| -ftree-coalesce-vars | -fgcse |  |
| -ftree-copy-prop 3 | -fhoist-adjacent-loads |  |
| -ftree-dce | -finline-small-functions |  |
| -ftree-dominator-opts | -findirect-inlining |  |
| -ftree-dse | -flra-remat |  |
| -ftree-forwprop | -foptimize-sibling-calls |  |
| -ftree-fre | -fpartial-inlining |  |
| -ftree-phiprop | -fpeephole2 |  |
| -ftree-sink | -ftree-tail-merge |  |
| -ftree-slsr | -fschedule-insns |  |
| -ftree-sra | -ftree-builtin-call-dce |  |
| -ftree-pta | -fipa-ra |  |
| -ftree-ter | -ftree-vrp |  |
| -funit-at-a-time | -ftree-pre |  |
| -fomit-frame-pointer | -foptimize-strlen |  |