

Disciplina: CIC 113468 – Introdução aos Sistemas Computacionais

Turma: A

Prof. Marcus Vinicius Lamar

Semestre: 2016/1

Lista de Exercícios 3

Assembly MIPS

Grupo:

Nome 1 Luis Felipe Braga Gebrim Silva Matrícula 1 16/0071569

Nome 2 Matrícula 2

Nome 3 Matrícula 3

Brasília

Junho de 2016

1. Dado o mapa de memória de código ao lado decodifique para Assembly MIPS.

.data

ENTRADA: .word 12

SAIDA: .word 0

1. Comente linha a linha o programa Assembly e diga a sua funcionalidade.

|  |  |  |
| --- | --- | --- |
| Mapa de Memória | Assembly | Comentário |
| 00400000 : 3c011001; | lui $at,0x1001 | Coloca o valor 0x1001 no registrador $at |
| 00400004 : 34300000; | ori $s0,$at,0 | Coloca o valor de $at em $s0 |
| 00400008 : 8e040000; | lw $a0,0($s0) | Carrega o valor armazenado no endereço $s0+0(ENTRADA) em $a0 |
| 0040000c : 0c100006; | jal l1 | Pula para o endereço 0x00400018 |
| 00400010 : ae020004; | sw $v0,4($s0) | Guarda o valor de $v0 no endereço $s0+4(SAIDA) |
| 00400014 : 08100014; | l2: j fim | Pula para o endereço 0x00400050 |
| 00400018 : 30880001; | l1: andi $t0,$a0,1 | Verifica se $a0 é impar, caso seja $t0 recebe 1, caso contrario $t0 recebe 0 |
| 0040001c : 1100000a; | beq $t0,$zero,l3 | Pula para o endereço 0x00400048 caso $t0 seja 0 |
| 00400020 : 00045842; | srl $t3,$a0,1 | Divide $a0 por 2 e coloca o resultado em $t3 |
| 00400024 : 24080003; | addiu $t0,$zero,3 | Coloca o valor 3 em $t0 |
| 00400028 : 0088001b; | l5: divu $a0,$t0 | Divide $a0 por $t0 |
| 0040002c : 00004810; | mfhi $t1 | Coloca o resto da divisão de $a0 por $t0 no registrador $t1 |
| 00400030 : 11200005; | beq $t1,$zero,l3 | Pula para o endereço 0x00400048 caso o valor de $t1 seja igual a 0 |
| 00400034 : 25080002; | addiu $t0,$t0,2 | Soma 2 ao valor de $t0 e coloca o resultado em $t0 |
| 00400038 : 010b602b; | sltu $t4,$t0,$t3 | Verifica se o valor de $t0 é menor que o de $t3, caso seja $t4 recebe 1, caso contrario recebe 0 |
| 0040003c : 1580fffa; | bne $t4,$zero,l5 | Pula para o endereço 0x00400028 caso o valor de $t4 seja 0 |
| 00400040 : 24020001; | addiu $v0,$zero,1 | Coloca o valor 1 no registrador $v0 |
| 00400044 : 08100013; | j l6 | Pula para o endereço 0x0040004c |
| 00400048 : 24020000; | l3: addiu $v0,$zero,0 | Coloca o valor 0 no registrador $v0 |
| 0040004c : 03e00008; | l6: jr $ra | Pula para o endereço indicado pelo registrador $ra |

Explique a funcionalidade do Programa:

Verifica se um número inteiro maior ou igual a 4 é primo, caso seja a saída é 1, caso contrário a saída é 0.

1. Escreva a saída para as seguintes entradas.

|  |  |
| --- | --- |
| Entrada | Saída |
| 5 | 1 |
| 12 | 0 |
| 737 | 0 |
| 7894 | 0 |
| 28291 | 0 |
| 2909813 | 1 |
| 379872301 | 0 |
| 4794987266 | 0 |

1. Quantos números entre 2 e 109+7 tem saída 1?

50.847.532

1. Qual é o maior número sem sinal que pode ser colocado na entrada (<232) e que a saída é 1?

4.294.967.291