

## Lista de Exercícios 2 – Parte 2

24)

Você deverá criar duas funções nesse exercício. Uma função que receba como argumentos 2 números inteiros de 32 bits. Essa função deverá também retornar um inteiro. O primeiro número recebido como parâmetro representa um endereço de memória e o segundo uma quantidade de elementos. A quantidade de elementos máxima é 30, se o numero recebido for superior a 30 sua função deverá usar 30. Os dois números acima deverão estar nas duas primeiras posições livres da memória (portanto devem ser lidos da memória para serem passados à função).

Sua função deverá criar um vetor que tem início no endereço de memória recebido como primeiro argumento e com a quantidade de elementos recebida como o segundo argumento.

Uma segunda função que receba um número (este número terá no máximo 16 bits) e retorne o seu quadrado.

Cada elemento do vetor y será dado como:

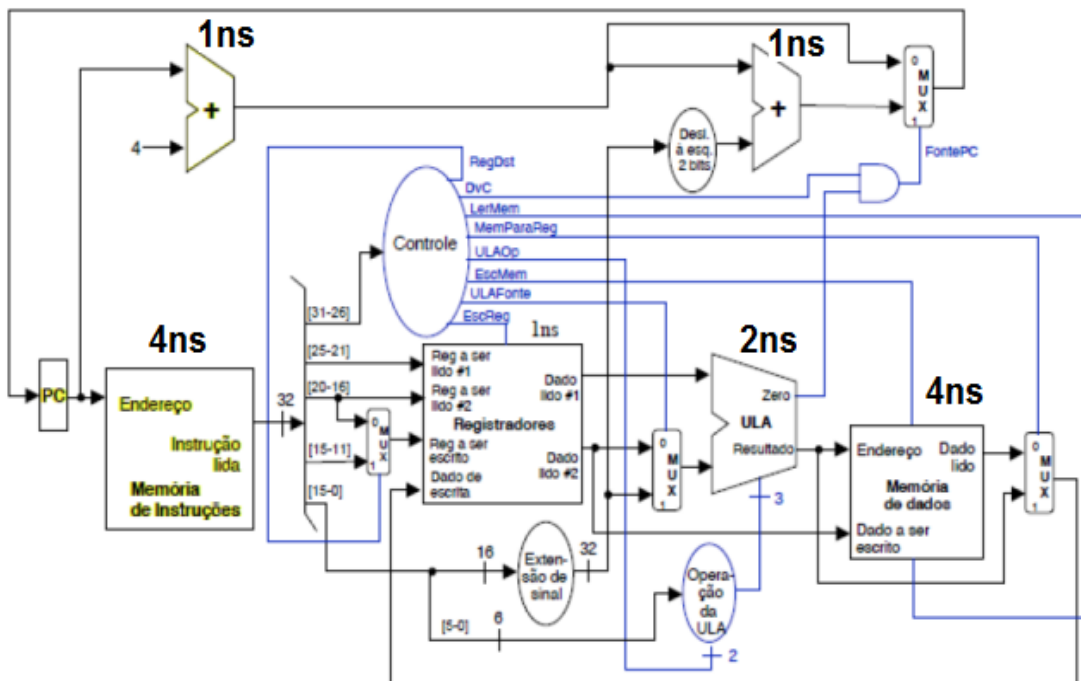
$y[i] = 2i^2 + 2i + 1$  se i for par;

$y[i] = i^2$  se i for impar.

O valor retornado será a soma de todos os elementos de y[].

25)

Considere o caminho de dados a seguir:



Através das instruções a seguir, explique as ações de cada unidade funcional do diagrama acima e como o controle atua em cada unidade.

- LW \$S1, num(\$S2)
- SW \$S1, num(\$S2)
- BEQ \$S1, \$S2, pulos
- ADD \$S1, \$S2, \$S3

**26)**

No diagrama está apresentado o tempo necessário para que cada unidade funcional execute a sua tarefa (para as unidades em que esse número não está presente, considere zero).

- a) Qual o tempo de execução de cada uma das instruções acima? (lembre-se que algumas coisas serão realizadas simultaneamente!)
- b) Compare os seguintes benchmarks considerando uma máquina monociclo e outra multiciclo (dê o speedup aproximado):
  - GCC (22% lw, 11% sw, 49% alu, 16% beq, 2% j)
  - ABC (11% lw, 49% sw, 22% alu, 2% beq, 16% j)

**27)**

Quais são as fases do pipeline do MIPS ? Divida o diagrama acima identificando essas fases.