

Infraestrutura de Hardware – EC

Lista de Exercícios – Capítulo 2 e 3 - continuação

Observações:

1. Coloquem as questões em um arquivo .zip e renomeie este arquivo .zip com o seu login do representante do grupo. Também deve ser enviado um arquivo pdf com o nome dos integrantes da equipe.
2. Façam códigos legíveis (indentados e comentados).
3. Usem o assembler do RISC-V e simulador CompSim
4. A lista pode ser feita em grupo, mas cópias entre equipes serão penalizadas

QUESTÕES

1. [1.5] Neste exercício você vai criar um jogo, descrito a seguir. O jogo a ser criado é em duplas e começa quando os jogadores 1 e 2 enviam uma palavra (string) cada um pelo teclado. Após enviar, o jogador espera pelo cálculo de sua pontuação que deverá ser calculado como descrito abaixo e mostrado na saída de vídeo. O jogador que obtiver maior pontuação será o vencedor. O programa deverá informar qual jogador ganhou ou se houve empate na saída de vídeo.

A pontuação de cada jogador deverá ser calculada como a soma dos valores referentes as letras de acordo com a tabela a seguir.

Letras	Valor
A, E, I, O, U	3
D, G, T	1
B, C, M, N, P	4
F, H, V, W, Y	2
K, R, S	5
J, L, X	8
Q, Z	6

Exemplo:

A pontuação da palavra "GOTICO" é 12, pois:

- 1 pontos para G
- 3 ponto para O, duas vezes
- 1 pontos para T
- 3 ponto para I
- 4 pontos para C

Obs: O jogo só aceitará **letras maiúsculas**.

Obs²: Acentuação será desconsiderada (não serão enviadas palavras com acentuação ou cedilha).

3. Implemente em Assembly um código que realize a multiplicação de dois números,

onde cada número terá no máximo 5 dígitos (use o algoritmo baseado em deslocamento).

- a. [1,0] Considere a situação em que os números estão na memória. Qual o tempo de execução máximo (em ciclos de clock) para uma plataforma baseada no processador RISC-V cycle-accurate?
 - b. [1,0] Agora implemente a multiplicação considerando que os números serão fornecidos usando o teclado e o resultado da multiplicação deverá ser visualizado na saída de vídeo. Calcule o tempo de execução (máximo) em ciclos de clock para uma plataforma baseada no processador RISC-V cycle-accurate.
4. [1.5] Este exercício tem como objetivo testar suas habilidades matemáticas com fatoriais. Para dificultar o teste, no exercício deverão ser somados os valores do fatorial de cada dígito do número. Escreva um código em Assembly do RISC-V que implementa esse exercício e permite a leitura de um número de até 5 dígitos e calcula a soma dos fatoriais dos dígitos exibindo a mesma na saída de vídeo.
- Ex 1.:

a. Entrada => 1234
Saída => 33 (1! + 2! + 3! + 4! = 33)

Ex 2.: Entrada => 678
Saída => 46080 (6! + 7! + 8! = 46080)

5. [1.5] Escreva um código em assembly do RISC-V que faça uma leitura analógica do sensor de temperatura do Arduino virtual e acenda um led verde, amarelo ou vermelho dependendo do valor da leitura conforme a tabela abaixo. Calcule o tempo de execução e CPI considerando um RISC-V cycle-accurate. Qual a taxa de leitura da sua solução? Se a taxa de leitura tiver que ter um speed-up de 10x, qual deverá ser a frequência do clock? É possível de implementar com a sua implementação?

Temperatura	Led
0-15	Amarelo -esquerda
16- 20	Verde – esquerda
21 - 25	Verde – direita
26 - 30	Amarelo – direita
31 – 35	Vermelho – direita
Maior que 35	Vermelho – esquerda

6. [1.5] Escreva um código que implementa um decodificador para o display de 7 segmentos. A partir do código de 4 bits colocado no teclado aparecerá o respectivo número no display de 7-segmentos do Arduino virtual. O número que aparecerá no display de 7 segmentos será de acordo com a entrada no teclado conforme a tabela

abaixo. Calcule o CPI para a implementação realizada considerando uma plataforma baseada no processador RISC-V cycle-accurate.

Entrada digital no teclado	Número do display 7 segmentos
0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7
1000	8
1001	9
1111	Limpa display