

Nota: Resolva este teste na folha adicional que lhe é fornecida para esse efeito.

Não esquecer de preencher nessa folha o número e nome.

3V

1. Usando o código apresentado abaixo, identifique para cada ciclo do relógio a ocupação de cada estágio do processador, para a versão PIPE- do Y86 – versão sem atalhos (preencha a tabela anexa). Justifique devidamente a sua resposta, indicando o porquê das bolhas injectadas.

```
I1: irmovl %10, %eax
I2: pushl %eax
I3: irmovl %20, %ebx
I4: popl %ecx
I5: addl %ebx, %ecx
```

Diagrama de ocupação dos estágios do processador (PIPE- Y86) para o código acima. As bolhas (bolhas) são indicadas nos estágios de execução de cada instrução. As setas indicam o fluxo de dados entre as instruções: %eax para %ecx, %ecx para %ebx, e %ebx para %ecx.

3V

2. Para o mesmo código do exercício 1, identifique para cada ciclo do relógio a ocupação de cada estágio do processador, para a versão PIPE do Y86 – versão com atalhos (preencha a tabela anexa). Justifique devidamente a sua resposta, indicando o porquê das bolhas injectadas ou sinais de atalho utilizados.

3. Apresenta-se abaixo uma sequência de operações elementares correspondentes a um ciclo:

```
ciclo:  load 0(%ebx, %eax.0, 4)  →  t.1
        addl t.1, %ecx.0        →  %ecx.1
        addl $1, %eax.0        →  %eax.1
        cmpl %eax.1, $1000     →  cc.1
        jl-taken cc.1, ciclo
```

2V
2V

- a. Transforme este código aplicando a técnica de loop splitting com grau 2:
b. Suponha que o código gerado na alínea anterior é executado num processador com as unidades funcionais, descritas na tabela abaixo. Apresente o diagrama de ocupação das unidades funcionais para as 2 primeiras iterações do ciclo.

| Unidade | Latência | Issue Time |
|-----------------------------|----------|------------|
| Operações inteiros | 1 | 1 |
| Operações inteiros + saltos | 1 | 1 |
| Leitura de memória (load) | 2 | 1 |