

1. Considere que a arquitetura Y86 suporta uma instrução *ipush C* a qual coloca a constante *C* no topo da pilha. Indique os valores dos sinais de controlo relevantes da organização sequencial do Y86 ativos em cada fase de execução da instrução. Que alterações são necessárias ao “datapath” do Y86 para suportar esta instrução.

2. Considere o seguinte programa em Y86. Identifique para cada ciclo do relógio a ocupação de cada estágio do processador, para a versão PIPE do Y86. Justifique a sua resposta e indique quais os valores encaminhados.

```

i1:    imovl $500,%ebx
i2:    xorl %eax, %eax
i3:    jne i6
i4:    addl ecx,%edx
i5:    halt
i6:    movl 1000(%ebx),%ecx
i7:    addl ecx,%edx
i8:    halt

```

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	Justificação

Nome: \_\_\_\_\_ Número: \_\_\_\_\_

3. Considere o seguinte extrato de programa escrito em Y86:

```
I1:    mrmovl 0(%ecx), %edx
I2:    addl  edx, %ebx
I3:    mrmovl 4(%ecx), %edx
I4:    addl  edx, %eax
I5:    addl  ebx, %eax
```

3.1 Apresente um grafo das dependências de dados existentes neste programa. Classifique cada uma das dependências existentes no programa.

3.2 Indique quais dessas dependências originam bolhas na arquitetura Y86 PIPE e explique como é que o escalonamento dinâmico e a renomeação de registos permitem remover essas bolhas.

Nome: \_\_\_\_\_ Número: \_\_\_\_\_

4. Considere o seguinte programa em C que multiplica todos os elementos de um vetor por uma constante alfa.

```
void apx(int v[], int N, int alfa) {  
    for(int i=0, i<N; i++)  
        v[i] = alfa * v[i];  
}
```

- 4.1 Apresente um programa em C resultante da aplicação da optimização designada por *desdobramento de ciclos* e considerando um desdobramento de 4 vezes. Assuma que o tamanho do vetor é múltiplo de 4. Quais os ganhos proporcionados por esta optimização?

- 4.2 Considere que pretende executar este programa numa arquitetura vetorial, designadamente, utilizando instruções SSE (as quais suportam registos de 128 bit, ou seja, 4 elementos). Indique, justificando, se esta versão seria mais rápida do que a desenvolvida na alínea anterior.