

Lista de Exercícios No. 02

- 1) Traduza o código a seguir para uma representação de três endereços assumindo que os valores inteiros requerem 4 bytes de armazenamento. Identifique os blocos básicos e o GFC.

```
if (i=2; i<=n; i++)
    a[i]=a[i]*10;
count=0;
s=n;
for (i=2; i<=s; i++)
    if (a[i]>100){
        count++;
        for (j=2*i; j<=n; j=j+i)
            a[j]=a[j]-10;
    }
```

- 2) Considere a existência de uma árvore sintática para o código a seguir. Mostre quais são as instruções de máquina para este código tendo como base o algoritmo para seleção de instruções baseado em matching de árvores.

```
t=a+b
u=t+c
v=u/d
```

- 3) Execute o algoritmo de programação dinâmica para as expressões a seguir. Considere a existência de três registradores e as instruções de máquina definidas no exercício anterior.

```
a+b*(c*(d+e))
(a+p)*((b-q)/(c+r))
```

- 4) Considere o programa P a seguir e utilize o algoritmo de list scheduling para definir um escalonamento com menor quantidade de ciclos possível. Considere que as operações de multiplicação consomem 2 ciclos e as demais operações 1 ciclo. Considere duas configurações de processadores. No processador P1, existe apenas uma ULA genérica que computa todas as operações. No processador P2, existem dois multiplicadores e duas ULAs, sendo que essas computam operações de soma, subtração e deslocamento.

```
f=2*a
g=b*c
j=f*g
m=j-f
t=m<<c
```

- 5) Otimize o código a seguir aplicando todas as otimizações possíveis (CSE, propagação de cópias, DCE, redução em força e eliminação de variáveis de indução). Mostre o código final e aponte os ganhos em termos de ciclos de relógio reduzidos com as otimizações. Considere que uma operação de multiplicação necessita de 3 ciclos e as demais instruções são executadas em 1 ciclo.

```
dp=0
i=0
n=10
L: t1=i*8
t2=A[t1]
t3=i*8
t4=B[t3]
t5=t2*t4
dp=dp+t5
i=i+1
if i<n goto L
```