1) Pipeline é uma técnica de implementação de processadores que permite a sobreposição temporal de diversas fases de execução de instruções, ou seja, o hardware executa mais de uma instrução de cada vez, sem esperar que uma instrução termine antes de iniciar a outra !

Vantagens:

- Melhor taxa de execução
- Acesso ao Memórias pode ser melhor utilizado dos estágios
- É uma técnica "invisível" aos programadores

Desvantagens:

- Risco de conflitos de dados
- Maior complexidade
- Controlar o tempo em cada estágio

2)

Instr.	Busca	Decod.	Exec.	Escrita
Instr. 1	200 ps	150 ps	300 ps	100 ps
Instr. 2	200 ps	150 ps	300 ps	100 ps

a) Sem pipeline: 750 ps
Com pipeline: 1250 ps

b) Sem pipeline: 98 (instruções) x 5 (estágios) = 1250 ps
Com pipeline: 1250 ps

c) Dividir o estágio de Exec em 2 estágios de 150 ps cada, então temos, o tempo de ciclo do clock passará a ser 200 ps !

d) Load 30% e Store 10%, totalizando 40% de utilização !

e) Desvio 20% e Load 30%, totalizando 50% de utilização !

3)

```
add x10,x11,x12
sub x13,x14,x15
lw x16,0(x17)
sw x18,0(x19)
add x20,x21,x22
sub x23,x24,x25
lw x26,0(x27)
sw x28,0(x29)
add x30,x31,x0
sub x1,x2,x3
lw x4,0(x5)
sw x6,0(x7)
add x8,x9,x10
sub x11,x12,x13
lw x14,0(x15)
sw x16,0(x17)
add x18,x19,x20
sub x21,x22,x23
lw x24,0(x25)
sw x26,0(x27)
add x28,x29,x30
sub x31,x0,x1
lw x2,0(x3)
sw x4,0(x5)
```

a) Diagrama de execução (ciclo x Instrução) :

b) $CicloS = 16 \times 12 + 4 = 196$
Instruções = 16 x 6 = 96
$\Rightarrow CPI = \frac{196}{96} = \boxed{2,04}$

c) Loop : lw x1, 0(x2)
sw x1, 0(x3)
add x1,x1,x1
sub x1,x1,x1
add x1,x1,x1
sub x1,x1,x1
lw x1,x1,-1
sub x1,x1,x1
sub x1,x1,x1
bne x1,x1,loop
add
add

$CicloS = 16 \times 10 + 9 = 169$
Instruções = 16 x 6 = 96
$\Rightarrow CPI = \frac{169}{96} = \boxed{1,76}$

d) Loop : lw x1, 0(x2)
lw x1, 0(x3)
add
add x1,x1,x1
sub x1,x1,x1
add x1,x1,x1
sub x1,x1,x1
lw x1,x1,-1
sub x1,x1,x1
bne x1,x1,loop
add
add

$CicloS = 16 \times 8 + 9 + 1 = 138$
Instruções = 16 x 6 = 96
$\Rightarrow CPI = \frac{138}{96} = \boxed{1,44}$

4)

a) $CicloS = 15 + 4 = \boxed{19}$ ∴ Tempo de Execução → $\boxed{19 \text{ ciclos}}$

b) Não é possível, pois a ULA calculando pelo endereço não pode ser utilizada nem pelos o acesso ao programa, já que um resultado depende do anterior !

c) Não haverá a desvantagem no estágio, não será calculado ULA para resolver conflitos de instruções que acessam a memória !

d)

Ciclo	Instr. 1	Instr. 2
1	Bs	Bs
2	Ds	Ds
3	Ex	Ex
4	Ex	Ex
5	Ex	Ex
6	Ex	Ex
7	Ex	Ex
8	Ex	Ex
9	Ex	Ex
10	Ex	Ex
11	Ex	Ex
12	Ex	Ex
13	Ex	Ex
14	Ex	Ex
15	Ex	Ex
16	Ex	Ex
17	Ex	Ex
18	Ex	Ex
19	Ex	Ex

5)

A) Sem forwarding → 100 ps ∴ 0,4 x 100 instruções (400) + 0 ciclos
Com forwarding → 100 ps ∴ 0,4 x 100 instruções (400) + 0 ciclos

$CPI_{sem} \Rightarrow \frac{ciclos}{instruções} \Rightarrow \frac{400 + 0}{400} \Rightarrow \boxed{1,0}$ ∴ $CPU_{sem} = 1 \times 0,5 \text{ ns} = \boxed{0,5 \text{ ns}} \Rightarrow \boxed{100 \times 200}$

$CPI_{com} \Rightarrow \frac{ciclos}{instruções} \Rightarrow \frac{400 + 0 \times 0}{400} \Rightarrow \boxed{1,0}$ ∴ $CPU_{com} = 1 \times 0,5 \text{ ns} \Rightarrow \boxed{0,5 \text{ ns}} \Rightarrow \boxed{100 \times 200}$

$\frac{CPU_{sem}}{CPU_{com}} \Rightarrow \frac{100 \times 200}{100 \times 200} \Rightarrow \boxed{1,0}$

$1,0 \Rightarrow \frac{100 \times 200}{100 \times 200} \Rightarrow \frac{1,0}{1,0} \Rightarrow \boxed{1,0}$

$1,0 \Rightarrow \boxed{1,0}$ "Nenhum ganho por speed up"

B) 0,4 → 100% → $CPI = 100 \times 200$
0,1 → 100% → $CPI = 0,6 \times 100 \times 200$

$Speed\text{-}Up = \frac{100 \times 200}{0,6 \times 100 \times 200}$ ou 1 $\Rightarrow$ $\frac{100 \times 200}{0,6 \times 100 \times 200} = \frac{1}{0,6}$
$\frac{100 \times 200}{0,6 \times 100 \times 200} = \frac{1}{0,6}$
$\frac{100 \times 200}{0,6 \times 100 \times 200} = \frac{1}{0,6}$
$\frac{100 \times 200}{0,6 \times 100 \times 200} = \frac{1}{0,6}$

$1 = \frac{1}{x}$
$\frac{1}{x} = \frac{1}{0,6}$
$x = \frac{1}{\frac{1}{0,6}}$
$x = \frac{0,6}{1}$
$x = 0,6$

$\frac{CPU_{sem}}{CPU_{com}} = \left(\frac{0,5}{0,1}\right) \Rightarrow \frac{1}{0,6} = \boxed{1,6}$
$\frac{CPU_{sem}}{CPU_{com}} = \left(\frac{0,5}{0,1}\right) \Rightarrow \frac{1}{0,6} = \boxed{0,833}$
∴ "Decrescimo valores de X"

C) $Speed\text{-}Up = \frac{Sem(Ciclos) + Sem}{Com(Ciclos) + Com}$ ou 1 $\Rightarrow$ $Sem(Ciclos) + Sem = Com(Ciclos) + Com$
$Sem + Sem = Com(Ciclos) + Com$
$Sem + Sem = Com(Ciclos) + Com$
$Sem + Sem = \frac{Com(Ciclos) + Com}{2}$

$Sem(Ciclos) + Sem = Com(Ciclos) + Com$
$Sem + Sem = Com(Ciclos) + Com$
$Sem + Sem = \frac{Com(Ciclos) + Com}{2}$

$Sem + Sem = \frac{Com(Ciclos) + Com}{2}$
$Sem + Sem = \frac{Com(Ciclos) + Com}{2}$
$Sem + Sem = \frac{Com(Ciclos) + Com}{2}$

$\frac{Sem + Sem}{Com} = \frac{Com(Ciclos) + Com}{2 \times Com}$
$\frac{Sem + Sem}{Com} = \frac{Com(Ciclos) + Com}{2 \times Com}$
$\frac{Sem + Sem}{Com} = \frac{Com(Ciclos) + Com}{2 \times Com}$

$\frac{Sem + Sem}{Com} = \frac{Com(Ciclos)}{2 \times Com} + \frac{Com}{2 \times Com}$
$\frac{Sem + Sem}{Com} = \frac{Com(Ciclos)}{2 \times Com} + \frac{Com}{2 \times Com}$
$\frac{Sem + Sem}{Com} = \frac{Com(Ciclos)}{2 \times Com} + \frac{Com}{2 \times Com}$

$\frac{Sem + Sem}{Com} = \frac{Com(Ciclos)}{2 \times Com} + \frac{1}{2}$
$\frac{Sem + Sem}{Com} = \frac{Com(Ciclos)}{2 \times Com} + \frac{1}{2}$
$\frac{Sem + Sem}{Com} = \frac{Com(Ciclos)}{2 \times Com} + \frac{1}{2}$

$\frac{Sem + Sem}{Com} - \frac{1}{2} = \frac{Com(Ciclos)}{2 \times Com}$
$\frac{Sem + Sem}{Com} - \frac{1}{2} = \frac{Com(Ciclos)}{2 \times Com}$
$\frac{Sem + Sem}{Com} - \frac{1}{2} = \frac{Com(Ciclos)}{2 \times Com