Universidade Federal da Fronteira Sul
Curso de Ciência da Computação
Disciplina: Organização de Computadores

1) Considerando ao função "fibonacci" apresentado abaixo:

a) Identifique as linhas que possuem conflitos de dados e descreva qual é a causa do conflito.

b) Identifique as linhas que possuem conflitos de controle e descreva a causa do conflito

c) Apresente o diagrama de tempo do pipeline para a execução da função “fibonacci” tendo em vista o valor recebido em a0. Considere que o hardware não resolve nenhum conflito de Pipeline e eles devem ser resolvidos em software inserindo bolhas no programa fonte antes da sua execução.

d) Apresente o diagrama de tempo do pipeline para a execução da função “fibonacci” tendo em vista o valor recebido em a0. Considere que o hardware resolve nos conflitos de Pipeline inserindo bolhas dinamicamente para os conflitos de dados e fazendo flush nos conflitos de controle.

e) Apresente o diagrama de tempo do pipeline para a execução da função “fibonacci” tendo em vista o valor recebido em a0 . Considere que o hardware resolve nos conflitos de Pipeline fazendo forwarding para os conflitos de dados e fazendo flush nos conflitos de controle.

f) Para cada um dos casos (c), (d) e (e) apresentados acima qual o total de instruções executadas e qual a quantidade de ciclos de clock necessárias deste o inicio da execução da função até a instrução ret terminar a sua execução.

```
01   .text
02
03 main:
04   li a0, 5
05   jal fibonacci
06   li  a7, 1
07   ecall
08   li  a7, 10
09   ecall
10 fibonacci:
11   addi t1, zero, 1 #fib1
12   addi t2, zero, 1 #fib2
13   addi t3, zero, 2
14   bgt  a0, t3, calcula
15   add  s0, zero, t1
16   ret
17 calcula:
18   beq  t3, a0, terminou
19   add  t4, t1, t2 #soma
20   add  t1, zero, t2
21   add  t2, zero, t4
22   addi t3, t3, 1
23   j    calcula
24 terminou:
25   add a0, zero, t4
26   ret
```

2)

A seguinte sequência de instruções lógicas e aritméticas será executada por um processador em *pipeline* de 5 estágios: busca da instrução, leitura de registradores, execução, acesso à memória e escrita de registradores. A sequência, no entanto, apresenta conflito de dados.

```
and   R5, R4, R3
or    R6, R4, R2
add   R1, R2, R2
mul   R3, R2, R1
sub   R1, R1, R4
```

O *pipeline* foi implementado sem *hardware* adicional para a resolução de conflitos, mas os valores dos registradores podem ser escritos na primeira metade do ciclo e lidos na segunda metade. Sabendo-se que o primeiro operando das instruções é o registrador destino, avalie as afirmações a seguir.

I. A troca de posição entre as instruções *or* e *add* soluciona o conflito de dados.

II. A troca de posição entre as instruções *add* e *and* soluciona o conflito de dados.

III. A inserção de uma operação *nop* (sem operação) entre *add* e *mul* soluciona o conflito de dados.

É correto o que se afirma em

A) I, apenas.
B) II, apenas.
C) I e III, apenas.
D) II e III, apenas.
E) I, II e III.