

- (1,5) 1. A Representação Intermédia (RI) tem como objectivo (escolha a resposta certa):
- A. Dar mais trabalho aos implementadores de compiladores
  - B. Obter uma representação abstracta mais próxima das operações da máquina final, mantendo alguma independência de plataforma
  - C. Facilitar a tradução directa para código assembly
  - D. Traduzir a APT para uma árvore mais próxima do código original, permitindo selecção de registos, variáveis e temporários.

- (1,5) 2. Um bloco básico é (escolha a resposta mais certa):
- A. Um bloco de código compreendido entre chavetas ({})
  - B. Uma sub-árvore da APT, contendo um LABEL e um JUMP
  - C. Uma sub-árvore da RI que tem apenas um caminho de código possível
  - D. Uma sub-árvore da RI que começa com um LABEL e termina com um JUMP

3. Considere o seguinte programa em *Ya!*:

```
1 f (n: int) : int {  
2   if n < 0 then {  
3     print ('Negativo ');  
4   }  
5   else {  
6     print ('Positivo ');  
7   };  
8 };  
9  
10 main () : void {  
11   f(1);  
12   f(-1);  
13 };  
14
```

- (3) (a) Proponha uma representação intermédia para a função `f()`.
- (1,5) (b) Proponha uma representação intermédia para a função `main()`.
- (3) (c) Quais são os blocos básicos presentes no código das alíneas anteriores? Quantos traços diferentes se podem gerar com esses blocos?
4. Aplicando as regras de reescrita para árvores canónicas, reescreva as seguintes árvores:
- (1,5) (a) `CALL(NAME(f), [CALL(NAME(g), [e1]), CALL(NAME(g), [e2]), CALL(NAME(h), [e3])])`
- (1,5) (b) `MOVE(TEMP(t), CALL(NAME(f), [e1, ESEQ(SEQ(s1, s2), e2), e3]))`
- (1,5) (c) `MOVE(MEM(ESEQ(s, e1)), e2)`
- (1,5) (d) `MOVE(e1, MEM(ESEQ(s, e2)))`
- (1,5) (e) `BINOP(PLUS, CALL(NAME(f), [TEMP(t)]), CALL(NAME(f), [ESEQ(MOVE(TEMP(t), CONST(0)), TEMP(t))]))`
- (2) 5. O `DISPLAY` permite o acesso a variáveis nos contextos lexicalmente superiores ao da função em execução, sendo imprescindível quando há declaração de funções dentro de funções (imbricadas).
- Sabendo que a linguagem *Ya!* não permite declaração de funções imbricadas, diga como poderíamos utilizar um `DISPLAY` na linguagem *Ya!*. Quantas células teria esse `DISPLAY`?