

# PARADIGMAS DE LINGUAGEM DE PROGRAMAÇÃO

## LISTA DE EXERCÍCIOS

1. Questões de projeto são indagações que precisam ser respondidas durante o projeto de uma linguagem de programação (LP). Em cada elemento que constitui uma LP há questões de projeto que levam os projetistas a uma decisão ou outra. Por exemplo, na definição do conjunto de caracteres que representará um espaço de memória. Quais são as questões de projeto que precisam ser avaliadas para o nome de uma variável?
2. Qual a relação entre uma palavra-chave e uma palavra reservada? O que difere uma da outra? De que forma palavras reservadas são melhores do que as palavras-chave?
3. No contexto de variáveis, o que é um *alias*? Cite exemplos em linguagens C e Java.
4. O que é o lado esquerdo de uma variável? O que é o lado direito?
5. Defina vinculação e tempo de vinculação.
6. Defina vinculação estática e vinculação dinâmica
7. Quais as vantagens e desvantagens de declarações implícitas?
8. Qual a diferença entre escopo e tempo de vida de uma variável?
9. O que é um bloco?
10. O que é ambiente de referenciamento?
11. Considere o seguinte programa de exemplo, em linguagem C:

```
void sub1() {  
    int a, b;  
    ... <----- 1  
} /* Fim de sub1 */  
  
void sub2() {  
    int b, c;  
    ... <----- 2  
    sub1();  
} /* end of sub2 */  
  
void main() {  
    int c, d;  
    ... <----- 3  
    sub2();  
} /* Fim de main */
```

Assuma que as únicas chamadas a funções são: main chama sub2 , que chama sub1. Quais são os ambientes de referenciamento em 1, 2 e 3, marcados no programa?

12. Considerando o mesmo trecho de código da questão anterior, quais os ambientes de referenciamento em 1, 2 e 3, marcados no programa, se a linguagem fosse de **escopo dinâmico**?

13. Considere o programa:

```
procedure Main is
  X, Y, Z : Integer;
  procedure Sub1 is
    A, Y, Z : Integer;
    procedure Sub2 is
      A, B, Z : Integer;
      begin -- de Sub2
        ...
      end; -- de Sub2
    begin -- de Sub1
      ...
    end; -- de Sub1
  procedure Sub3 is
    A, X, W : Integer;
    begin -- de Sub3
      ...
    end; -- de Sub3
  begin -- de Main
    ...
  end; -- de Main
```

Liste todas as variáveis, com as unidades de programa onde elas estão declaradas, visíveis nos corpos de Sub1, Sub2 e Sub3, assumindo que o escopo estático esteja sendo usado.

14. Considere o seguinte esqueleto de programa em C:

```
void fun1(void); /* protótipo */
void fun2(void); /* protótipo */
void fun3(void); /* protótipo */

void main(void) {
  int a, b, c;
  ...
}

void fun1(void) {
  int b, c, d;
  ...
}

void fun2(void) {
  int c, d, e;
  ...
}

void fun3(void) {
  int d, e, f;
  ...
}
```

Dada as seguintes sequências de chamadas e assumindo que o escopo dinâmico é usado, que variáveis são visíveis durante a execução da última função chamada? Inclua com cada variável visível o nome da função na qual ela foi definida.

a) main chama fun1; fun1 chama fun2; fun2 chama fun3.

- b) main chama fun1; fun1 chama fun3.
  - c) main chama fun2; fun2 chama fun3; fun3 chama fun1.
  - d) main chama fun3; fun3 chama fun1.
  - e) main chama fun1; fun1 chama fun3; fun3 chama fun2.
  - f) main chama fun3; fun3 chama fun2; fun2 chama fun1.
15. Quais argumentos são a favor e contra a representação de valores booleanos com bits únicos em memória?
  16. De que forma um valor do tipo decimal perde espaço em memória?
  17. Que justificativa significativa existe para o operador (->) em C e C++? Dica: no contexto em que deve-se usar o operador (->), é possível usar o operador . (ponto) também.
  18. Cite as diferenças entre os tipos de enumeração em C++ e em Java.
  19. As uniões em C e em C++ são separadas dos registros, diferentemente de Ada, que têm a definição dessas duas estruturas combinadas, com uma embutida na outra. Quais as vantagens e desvantagens dessas duas escolhas?
  20. Descreva o que foi perdido e o que foi ganho na decisão dos projetistas de Java de não incluírem os ponteiros de C++.
  21. O que é uma linguagem fracamente tipada e uma linguagem fortemente tipada? Quais vantagens e desvantagens de uma em relação à outra?
  22. Escreva um programa em C que realiza a operação de soma entre duas matrizes. (Observe quais restrições essa operação impõe)
  23. Escreva um programa em C que realiza a operação de multiplicação entre duas matrizes. (Observe quais restrições essa operação impõe)
  24. Reescreva o programa da questão 23 fazendo o uso de ponteiros para definição e manipulação de matrizes.
  25. Dado um vetor, **v**, de tamanho **N**, escreva em C um programa que escreve em um segundo vetor, **w**, uma fatia de **v**. A fatia deve ser dada pelas posições de **a** até **b-1**, de modo que **0 ≤ a < b ≤ N**. Dessa forma, a primeira posição de **w** armazenará o valor da posição dada por **a** de **v**. Por fim, imprima o segundo vetor – da posição **0** até a posição **b-1-a**.

Para os programas das questões 22, 23, 24 e 25, implemente todo o programa na função main. Os códigos dessas quatro questões deverão ser enviadas para o SIGAA, em tarefa aberta, até o dia 15/05/2018. Cada questão valerá 0,5 ponto, de modo que a prova valerá de 0,0 a 8,0. Não deixe para enviar no último dia, muito menos na última hora, pois por azar, é possível que o sistema esteja instável. Envio fora do prazo terá um desconto de 0,5 ponto por cada hora de atraso. A tarefa ficará aberta até o dia 16/05, no entanto, o prazo para envio será terça-feira, dia 15/05. Levando em consideração a tolerância, em hipótese alguma a tarefa será aceita fora do prazo.