Lista de Exercícios de Linguagens de Programação — Capítulo 2.
Primeiro Semestre de 2000. Dept. de Computação – UFSCar.
José de Oliveira Guimarães.

1. O que é um tipo ?

2. O que é tipagem estática e dinâmica ? Explique através de exemplos.

3. Explique a diferença entre linguagens fortemente tipadas e as estaticamente tipadas ?

4. Toda linguagem fortemente tipada é tipada dinamicamente ?

5. O que é um erro de tipos ? Cite um exemplo onde o erro é descoberto em compilação e outro onde ele só é apontado em tempo de execução.

6. Escreva um programa em uma linguagem dinamicamente tipada no qual um erro de tipos *pode* ou não acontecer durante a execução. Por exemplo, o programa pode ler um valor do teclado e, dependendo do valor, o erro acontece ou não.

7. Por que o código gerado para linguagens estaticamente tipadas é mais rápido do que o código das linguagens dinamicamente tipadas ?. Explique utilizando um exemplo.

8. Dê um exemplo de um programa em S que utilize procedimentos encaixados. O programa deve funcionar, mesmo que não faça nada de útil.

9. Cite as desvantagens de estruturas de blocos.

10. Faça um pequeno programa com dois procedimentos na linguagem S com escopo dinâmico de tal forma que :

- não haja erro de compilação nem de execução;
- se a linguagem suportasse apenas escopo estático, haveria um erro em tempo de compilação.

11. Admita que o programa abaixo seja de uma linguagem com escopo dinâmico. O que ele escreve ? Naturalmente, a execução do programa começa no procedimento `main`.

```
proc P()
  var i, n : integer;
  begin
  n = k + r;
  for i = 1 to n do
    max = max + i;
  end
```

```
proc Q( max : integer )
  var k : integer;
  begin
  k = 1;
  P();
  writeln(max);
  k = 3;
  P();
  writeln(max);
  writeln(n);
  end


proc main()
  var n, r : integer;
  begin
  n = 5;
  r = 1;
  Q(n);
  end
```

12. O que escreve o programa abaixo ? O que acontece se a constante `max` for inicializada com `1` ?

```
    { variaveis globais }
  var a, b;

proc P( i )
  begin
  if i > 3
  then
    a = 12;
    b = "abcdefg";
  else
    a = "abcdefg";
    b = 33;
  endif
  end

proc main()
  var s, j;
  begin
  max = 5;
  s = 1;
  for j = 1 to max do
    s = s + j;
  P(s);
  a = a + 1;
  writeln(a);
```

```
end
```

13. Explique como módulos funcionam. A sua resposta deve explicar o que é parte pública e privada e o que um módulo pode utilizar de outro módulo que ele importa.

14. Cite duas vantagens de se utilizar módulos.

15. Faça um pequeno exemplo com dois procedimentos de tal forma que uma exceção levantada em um deles seja tratada no outro. Utilizando este exemplo, explique como exceções funcionam.

16. Faça um exemplo onde uma exceção é levantada e não tratada.

17. Cite duas desvantagens de exceções.

18. Que tipos de erros podem acontecer quando o programador desaloca memória explicitamente ?

19. Cite quatro problemas com a desalocação explícita de memória pelo programador (`delete` de C++, `dispose` de Pascal).

20. Cite dois problemas com coleta de lixo (mesmo se os coletores atuais já resolveram estes problemas, pelo menos parcialmente).