**PARADIGMAS DE PROGRAMAÇÃO**

**EXERCÍCIOS LISTA 07**: **SUBPROGRAMAS**

Prof.ª Lucília Ribeiro

Aluno: **Thiago Santos Oliveira** Matrícula:20141002803464

**RESPOSTA 01:** As características gerais de subprogramas são:

-Cada subprograma tem um único ponto de entrada.

-A unidade de programa chamadora é suspensa durante a execução do subprograma chamado, implicando a existência de apenas um subprograma em execução em qualquer momento no tempo.

-O controle sempre retorna para o chamador quando a execução do subprograma termina.

**RESPOSTA 02:**

PARÂMETROS FORMAIS – Os parâmetros no cabeçalho do subprograma. São às vezes vistos como variáveis fictícias porque não são variáveis no sentido usual: na maioria dos casos, são vinculados ao armazenamento apenas quando o subprograma é chamado, e essa vinculação é normalmente feita por meio de alguma outra variável do programa.

PARÂMETROS REAIS – Sentenças de chamadas a subprogramas devem incluir o nome do subprograma e uma lista de parâmetros para serem vinculados aos pa

**RESPOSTA 03:** Questões de projeto para subprogramas:

-As variáveis locais são alocadas estaticamente ou dinamicamente?

-As definições de subprogramas pode aparecer em outras definições de subprogramas?

-Que método ou métodos de passagem de parâmetros são usados?

-Os tipos dos parâmetros reais são verificados em relação aos tipos dos parâmetros formais?

-Se os subprogramas puderem ser passados como parâmetros e puderem ser aninhados, qual é o ambiente de referenciamento de um subprograma passado como parâmetro?

-Os subprogramas podem ser sobrecarregados?

-Os subprogramas podem ser genéricos?

**RESPOSTA 04:** Em funções C++, as variáveis são dinâmicas da pilha (podem ser vinculadas ao armazenamento quando o subprograma começar a executar e desvinculadas quando a execução terminar), ao menos que sejam especificamente declaradas como estáticas (usando static). Logo para o método em questão, seria necessário o uso de variável estática, já que variáveis dinâmicas da pilha, o método não seria sensível ao histórico. Logo o método pode ser feito sem parâmetros e sem referências não locais, utilizando variáveis locais.

**RESPOSTA 05:**

A maior desvantagem das variáveis locais estáticas é sua inabilidade de suportar recursão. E uma das principais desvantagens das variáveis locais dinâmicas da pilha é que existe o custo em termos de tempo necessário para alocar, inicializar (quando necessário) e liberar as variáveis para cada chamada ao subprograma.

**RESPOSTA 06:**

1. Value = 2, list[5] = {1, 3, 5, 7, 9};
2. value = 1, list[0] = 2

list[0] = 3, list[1] = 2

value = 2, list[value] =1

1. NÃO ENTENDI A PASSAGEM POR RESULTADO

**RESPOSTA 07:**

1. List[2] = {1, 3}
2. List[2]= {2,6}
3. NÃO ENTENDI A PASSAGEM POR RESULTADO

**RESPOSTA 08:**

**RESPOSTA 09:**

**RESPOSTA 10:**