

Lógica de Programação e Algoritmos

Caros alunos, as videoaulas desta disciplina encontram-se no AVA
(Ambiente Virtual de Aprendizagem).



| Unidade 4

Sub-Rotinas

Introdução da Unidade

Olá, Querido Aluno! Seja muito bem-vindo à disciplina de Lógica de Programação e Algoritmos. Nesta quarta Unidade, veremos a modularização dos algoritmos implementadas em sub-rotinas. Na primeira aula iremos aprender sobre a decomposição de um problema em vários subproblemas, sendo cada um implementado em uma função ou procedimento. Na segunda aula iremos aprender sobre o escopo de variáveis, passagem de parâmetros e funções recursivas. As funções e procedimentos são implementadas em diversas linguagens de programação, por exemplo, em Java como métodos. Espero que você aproveite os conteúdos que serão explanados e faça as atividades propostas que visam fortalecer o seu entendimento sobre o tópico em questão. Sendo assim, vamos iniciar.

Objetivos

- Compreender o conceito de sub-rotinas e suas aplicações;
- Aprender sobre o escopo e uso de variáveis;
- Entender a passagem de parâmetros;
- Entender o funcionamento da função recursiva.

Conteúdo programático

Aula 01 – Função e Procedimento.

Aula 02 – Escopo de Variáveis, Passagem de Parâmetros e Recursividade.

Referências

ASCENCIO, Ana Fernanda Gomes; CAMPOS, Edilene Aparecida Veneruchi de. **Fundamentos da programação de computadores**: Algoritmos, Pascal, C/C++ e Java. 2. ed. São Paulo: Pearson, 2007.

FORBELLONE, André Luiz Villar; EBERSPÄCHER, Henri Frederico. **Lógica de programação**: A construção de algoritmos e estruturas de dados. 3. ed. São Paulo: Prentice Hall, 2005.



Você poderá também **assistir às videoaulas** em seu celular! Basta apontar a câmera para os **QRCodes** distribuídos neste conteúdo.

Pode ser necessário instalar um aplicativo de leitura QRcode no celular e efetuar login na sua conta Gmail.

Aula 1 Função e Procedimento.



Videoaula 1

Utilize o QR Code para assistir!

Inicialmente, assista ao vídeo que aborda sobre as sub-rotinas.



Decomposição

Ao decompor um problema em subproblemas, ocorre a divisão da complexidade e assim, simplifica a resolução, permitindo focar a atenção em um problema pequeno de cada vez, no final, a soma de todos os subproblemas resulta em uma melhor compreensão do problema.

É conveniente que adotemos, então, um critério para orientar o processo de decomposição (FORBELLONE; EBERSPÄCHER, 2005):

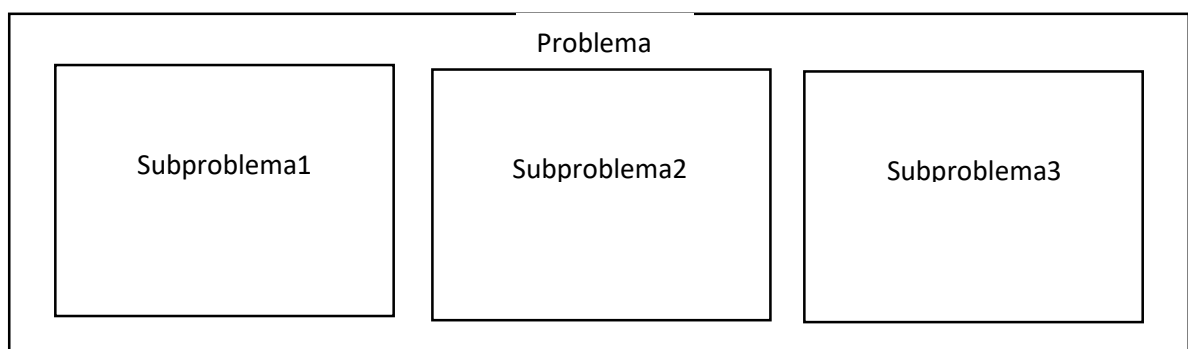
- Dividir o problema em suas partes principais;
- Analisar a divisão obtida para garantir coerência;
- Se alguma parte ainda permanecer complexa, decompô-la também;
- Analisar o resultado para garantir entendimento e coerência.

A decomposição também é conhecida como refinamentos sucessivos, parte de um problema complexo que é sucessivamente dividido até definir problemas mais simples.

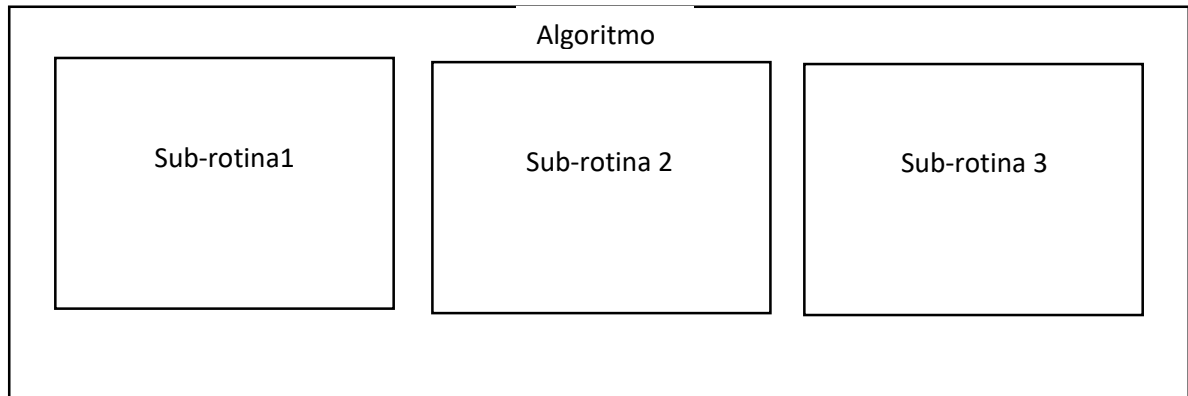
Módulos

Após decompor um problema em subproblemas, para cada subproblema deve ser escrito uma sub-rotina ou módulo.

Decomposição do problema:



Estrutura do algoritmo:

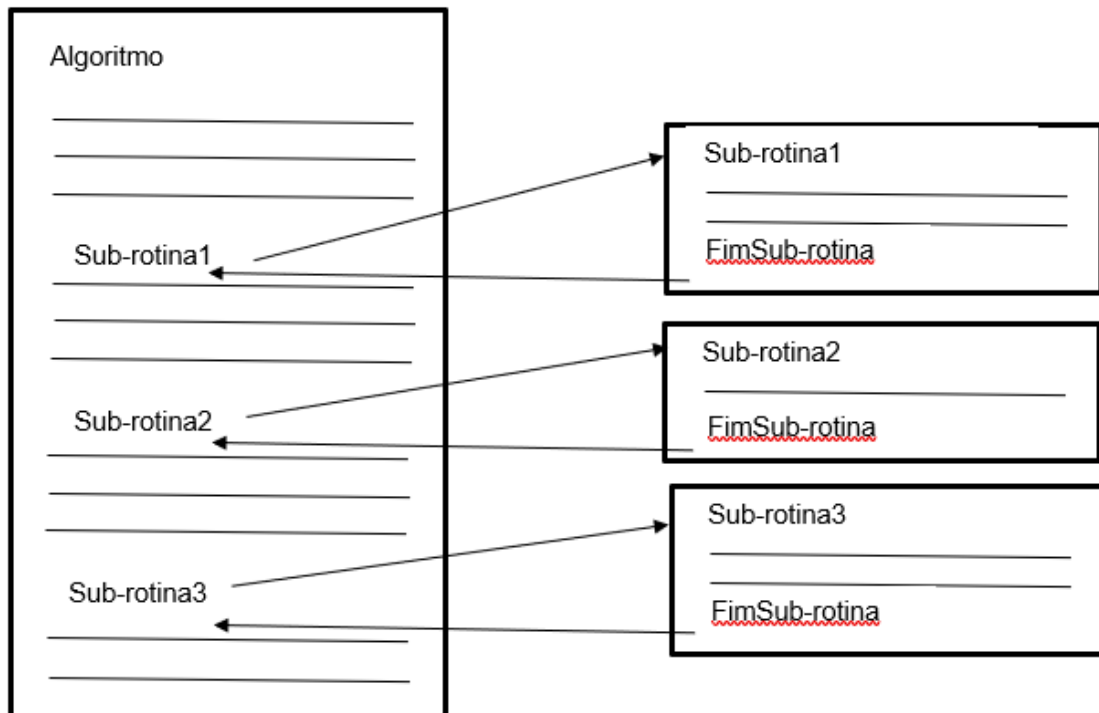


Ativação da sub-rotina

A ativação acontece quando um determinado ponto do algoritmo tem descrito o nome da sub-rotina, neste ponto ocorre a chamada da sub-rotina.

Na chamada da sub-rotina, o fluxo de execução é desviado para ela, após seu término, o fluxo de execução retorna ao algoritmo, no primeiro comando após a chamada.

Representação da ativação das sub-rotinas:





Videoaula 2

Utilize o QR Code para assistir!

Agora, assista ao vídeo que aborda sobre como criar e utilizar uma função.



Declaração da Função

Funcao <identificador> (p1, p2 : tipo) : tr

//declaração de variáveis

//comandos

Fimfuncao

Identificador é o nome da função que será referenciada no algoritmo.

p1 e p2 são parâmetros da função.

tr é tipo de dado que é retornado pela função.

Exercícios

1. Elabore um algoritmo contendo uma função que retorne POS se o número digitado for positivo ou NEG se for negativo.
2. Elabore um algoritmo contendo uma função que receba por parâmetro dois números inteiros e retorne a soma dos números inteiros existentes entre eles. Mostre o valor da soma.
3. Faça um algoritmo contendo uma função que receba por parâmetro três notas de um aluno e uma letra. Se a letra for A, a função deverá calcular a média aritmética das notas do aluno; se for P, deverá calcular a média ponderada, com pesos 1, 3 e 6. A média calculada deverá ser devolvida ao algoritmo principal para ser mostrada.



Videoaula 3

Utilize o QR Code para assistir!

Agora, assista ao vídeo que aborda sobre
criar e utilizar procedimentos



Declaração do Procedimento

Procedimento <identificador> (p1, p2 : tipo)

//declaração de variáveis

//comandos

Fimprocedimento

Identificador é o nome do procedimento que será referenciado no algoritmo.

p1 e p2 são parâmetros do procedimento.

Exercícios

1. Crie um algoritmo que receba os valores antigo e atual de um produto. Chame um procedimento que determine o percentual de acréscimo entre esses valores. Mostre o percentual ao final. Considere que o valor atual é sempre maior que o antigo.
2. Faça um algoritmo que contenha um procedimento que leia cinco números inteiros, determine e mostre o maior e o menor deles.
3. Faça um algoritmo que contenha um procedimento que faça a entrada de dados do vetor A de dez elementos inteiros. Faça um procedimento que gere um vetor B contendo o fatorial de cada elemento de A. Faça uma função para retornar o fatorial. O vetor B deverá ser mostrado no programa principal.

Aula 2 Escopo de Variáveis, Passagem de Parâmetros e Recursividade



Videoaula 1

Utilize o QR Code para assistir!

Agora, assista ao vídeo que aborda sobre o escopo de variáveis globais e locais.



Escopo de variáveis

O escopo ou abrangência de uma variável, na realidade, denota sua visibilidade (conhecimento e aplicação) perante os diversos módulos integrantes do algoritmo. A visibilidade é relativa à hierarquia; podemos dizer, então, que uma variável é global a todos os módulos hierarquicamente inferiores e é local quando é visível apenas em seu contexto e não aos módulos hierarquicamente superiores (FORBELLONE; EBERSPÄCHER, 2005).

Exemplo:

Algoritmo "Escopo"

Var

G1 : inteiro

procedimento subrot1(x : inteiro)

Var

L1 : inteiro

Inicio

escreval("G1:", G1)

L1 <- x

escreval("L1:", L1)

Fimprocedimento

Inicio

G1 <- 10

subrot1(50)

Fimalgoritmo

Observe que a variável "G1" definida no início do algoritmo é visível em todo o algoritmo, na sub-rotina subrot1 e em qualquer sub-rotina que seja declarada, ela é global a todos. Em outra situação, a variável "L1" é local na sub-rotina subrot1, não sendo visível no algoritmo principal.

Exercícios

1. Faça um algoritmo contendo uma função que receba dois valores numéricos e um símbolo. Este símbolo representará a operação que se deseja efetuar com os números. Se o símbolo for + deverá ser realizada uma adição, se for – deverá ser realizada uma subtração, se for * deverá ser efetuada uma multiplicação e se for / deverá ser realizada uma divisão. O resultado deverá ser mostrado no algoritmo principal.
2. Elabore uma sub-rotina que receba duas cadeias de caracteres como parâmetros e retorne 1 se elas forem iguais. Caso contrário, retorne 0.
3. Crie uma sub-rotina que receba como parâmetro a altura e o sexo de uma pessoa e retorne o seu peso ideal. Para homens, deverá calcular o peso ideal usando a fórmula: peso ideal = $72.7 * \text{altura} - 58$; para mulheres: peso ideal = $62.1 * \text{altura} - 44.7$



Videoaula 2

Utilize o QR Code para assistir!

Agora, assista ao vídeo que aborda sobre passagem de parâmetros.



Passagem de parâmetros

Uma sub-rotina tem duas formas para comunicar-se com o mundo externo:

- Por variáveis globais;
- Por passagem e retorno de valores.

No algoritmo abaixo, as variáveis globais numero1 e numero2 são somadas no procedimento soma, observe que o procedimento não possui parâmetros e a comunicação é através das variáveis globais, e a soma é dependente das duas variáveis.

Algoritmo "SomaGlobal"

Var

numero1, numero2 : inteiro

procedimento soma

Var

resultado : inteiro

Inicio

resultado <- numero1 + numero2

escreva("Soma:", resultado)

Fimprocedimento

Inicio

escreva("Informe o primeiro número:")

leia(numero1)

escreva("Informe o segundo número:")

leia(numero2)

soma

Fimalgoritmo

No algoritmo abaixo, o procedimento soma possui dois parâmetros inteiros que são utilizados na soma. Na primeira chamada do procedimento soma-se as variáveis globais, numero1 e numero2, que são passadas como parâmetros, na segunda chamada a variável global numero1 e o valor 10 são passados como parâmetros, e na terceira chamada, as variáveis globais numero2 e numero3. Observe que o procedimento com os parâmetros ficou genérico, realiza qualquer soma de dois valores inteiros sem depender de variáveis globais específicas.

Algoritmo "SomaParam"

Var

numero1, numero2, numero3 : inteiro

procedimento soma(num1, num2 : inteiro)

Var

resultado : inteiro

Inicio

resultado <- num1 + num2

```
escreval("Soma:", resultado)
```

Fimprocedimento

Inicio

```
escreva("Informe o primeiro número:")
```

```
leia(numero1)
```

```
escreva("Informe o segundo número:")
```

```
leia(numero2)
```

```
soma(numero1,numero2)
```

```
soma(numero1,10)
```

```
escreva("Informe o terceiro número:")
```

```
leia(numero3)
```

```
soma(numero2, numero3)
```

Fimalgoritmo

Exercícios

1. Foi realizada uma pesquisa sobre algumas características físicas de cinco habitantes de uma região. Foram coletados os seguintes dados de cada habitante: sexo, cor dos olhos (A – azuis ou C – castanhos), cor dos cabelos (L – loiros, P – pretos ou C – castanhos) e idade (ASCENCIO; CAMPOS, 2007).
 - a. Faça um procedimento que leia esses dados, armazenando-os em vetores;
 - b. Faça uma função que determine e devolva ao programa principal a média de idade das pessoas com olhos castanhos e cabelos pretos;
 - c. Faça uma função que determine e devolva ao programa principal a maior idade entre os habitantes;
 - d. Faça uma função que determine e devolva ao programa principal a quantidade de indivíduos do sexo feminino com idade entre 18 e 35 anos (inclusive) e que tenham olhos azuis e cabelos loiros.
2. Faça um algoritmo que contenha um procedimento que ordene os 5 números inteiros do vetor A de forma crescente.
3. Faça um algoritmo que tenha um procedimento com passagem de parâmetro por valor e outro procedimento com passagem de parâmetro por referência. Explique a diferença entre as duas formas de passagem de parâmetro.



Videoaula 3

Utilize o QR Code para assistir!

Agora, assista ao vídeo que aborda sobre recursividade.



Exercícios

1. Faça um algoritmo contendo uma função recursiva que receba como parâmetro um número inteiro positivo Num e calcule o somatório dos números de 1 a Num. Faça a chamada da função no trecho principal do algoritmo.
2. Faça um algoritmo contendo uma função recursiva para calcular a potência. Receba como parâmetros a base e o expoente. No trecho principal do algoritmo, faça a chamada da função.
3. Faça um algoritmo contendo uma função recursiva que receba como parâmetro um número inteiro Num e calcule a sequência de Fibonacci. Faça a chamada da função no trecho principal do algoritmo.

Encerramento

Chegamos ao final dos estudos da nossa quarta Unidade. Tivemos a oportunidade de entender sobre as sub-rotinas e suas aplicações. Nas aulas, nos dedicamos na compreensão das funções e procedimentos, escopo de variáveis locais e globais, passagem de parâmetros por valor e por referência para aumentar a generalidade da sub-rotina, e por fim, as funções recursivas, todo o conteúdo com apoio de exercícios visando o aprimoramento na lógica de programação.



Videoaula Encerramento

Utilize o QR Code para assistir!

Assista agora ao vídeo de encerramento de nossa disciplina.



Encerramento da Disciplina

Chegamos ao fim de mais uma disciplina do seu curso! Nosso objetivo durante toda a disciplina foi apresentar a base para iniciar seu aprendizado e aperfeiçoamento no desenvolvimento de programas de computador. Você deve ter percebido que a lógica de programação é essencial na solução de problemas computacionais e digo que o conhecimento adquirido aqui com os tipos de dados, variáveis, operadores, expressões, comandos de decisão e repetição, estruturas de dados e sub-rotinas, você poderá utilizá-los quando estiver aprendendo uma linguagem de programação, como por exemplo, Java. Com a inovação tecnológica surgirão novas linguagens que você irá interessar e aprender, mas a lógica de programação continuará sendo a sua base. Espero que você tenha tido um bom aproveitamento e compreendido os temas que foram expostos aqui. Te desejo muito sucesso e nos vemos em uma próxima oportunidade. Até lá!

Esperamos que este guia o tenha ajudado compreender a organização e o funcionamento de seu curso. Outras questões importantes relacionadas ao curso serão disponibilizadas pela coordenação.

Grande abraço e sucesso!

