

Algoritmo s e Técnicas de Programa

Prof. Me. Anderson E. Macedo

Conçalves



Aula 3: Recursividade



UNIDADE 4 – Funções e recursividade

Objetivo da aula

1.

Recursividade

Definições e conceitos da recursividade.

2.

Exemplos da Recursividade

Demonstração de exemplos que podem ser resolvidos com a recursividade.

3.

Realidade profissional

Aplicar exemplos da realidade profissional.

4.

Palavras-chave

Recursividade, iteração, subproblemas.

Recursividade

A recursividade é a capacidade de uma função chamar a si mesma durante sua execução. Algumas vantagens:

Resolver problemas complexos

Divide problemas em partes menores

Soluções mais claras

Modularidade ao código

Aplicabilidade da recursividade

Cálculo do fatorial de um número, por exemplo 5.

Fatorial = $(5 * 4 * 3 * 2 * 1)$

```
1  #include <stdio.h>
2
3  int fatorial(int n) {
4      // Caso base
5      if (n == 0 || n == 1) {
6          return 1;
7      }
8      // Caso recursivo
9      else {
10         return n * fatorial(n - 1);
11     }
12 }
13
14 int main() {
15     int numero = 5;
16     printf("O fatorial de %d é: %d\n", numero, fatorial(numero));
17     return 0;
18 }
```

Realidade Profissional



Revisando

1.

Recursividade

Definições e conceitos da recursividade.

2.

Exemplos da Recursividade

Demonstração de exemplos que podem ser resolvidos com a recursividade.

3.

Realidade profissional

Aplicar exemplos da realidade profissional.

4.

Palavras-chave

Recursividade, iteração, subproblemas.