

# Combinatória

Grupo 03/04

# Tópicos Abordados

- Técnica de contagem
- Relações de recorrência
- Coeficientes binomiais
- Sequências de contagem
- Recursão
- Indução

# Fatorial

Produto dos números naturais começando em **n** e decrescendo até **1** denominamos de **fatorial de n** e representamos por **n!**

**Pode ser definido recursivamente ou iterativamente**

$$n! = n * (n-1) * (n-2) * (n-3) * \dots * (n-(n-1))$$

$$n! = n * (n-1)!$$

$$5! = 5 * 4 * 3 * 2 * 1 = 120$$

$$5! = 5 * 4!$$

$$4! = 4 * 3!$$

$$3! = 3 * 2!$$

$$2! = 2 * 1!$$

$$1! = 1 * 0!$$

$$0! = 1$$

# Fatorial

```
#include<stdio.h>
int main (void)
{
    int num, fat=0, y;
    scanf("%d",&num);
    y=num-1;
    fat=num*y;
    y--;
    while(y>0){
        fat=fat*y;
        y--;
    }
}
```

# Fatorial Recursivo

```
#include <stdio.h>

int fat (n) {
    if ((n==1) || (n==0))
        return 1;
    else
        return fat(n-1)*n;
}
```

# Arranjo

Fórmula:  $A(n, p) = n! / (n - p)!$

Utilidade: É muito útil na solução de problemas de contagem onde a **ordem** é levada em consideração.

# Combinação

Fórmula:

- $C_{n,p} = \frac{n!}{p! (n - p)!}$
- Utilidade: É muito útil na solução de problemas de contagem onde a **ordem não** é levada em consideração.

# MDC

Maior divisor comum de dois ou mais números

$$\text{mdc}(6,12) = 6$$

$$\text{mdc}(12,20) = 4$$

$$\text{mdc}(20,24) = 4$$

$$\text{mdc}(12,20,24) = 4$$

$$\text{mdc}(6,12,15) = 3$$



# MDC

Para calcular o m.d.c.de 33 e 48:

1) Divide-se o maior número pelo outro.

$48 \div 33$  dá 1, resto 15

O resto não é zero.

2) O divisor passa a dividendo, o resto passa a divisor e divide-se de novo.

$33 \div 15$  dá 2, resto 3

3) Como o resto não é zero, repete-se novamente 2)

$15 \div 3$  dá 5, resto 0

Agora, a divisão não deixou resto; e o último resto diferente de 0 é o máximo divisor comum.

# MDC – Abordagem iterativa

```
int MDC(int a,int b){  
  
    int menor;  
    if (a>b)    menor =b;  
    else    menor =a;  
  
    for (int i=menor; i>=1; i--)  
        if (a%i==0 && b%i==0)  
  
        return i;  
  
}
```

# MDC – Abordagem recursiva

```
int MDC (int a, int b) {  
  
    if (b==0) return a;  
  
    return MDC (b, a%b) ;  
}
```

# MMC

O mmc pode ser definido utilizando a definição de MDC:

$$\text{mmc}(a, b) = \frac{a \cdot b}{\text{mdc}(a, b)}.$$

```
int MMC(int a,int b){  
    return a*b/MDC(a,b);  
}
```

# Primos

**Números primos** são os números naturais que têm **apenas dois divisores diferentes**: o 1 e ele mesmo.

## **Exemplos:**

- 1) **2** tem apenas os divisores **1** e **2**, portanto **2** é um número primo.
- 2) **17** tem apenas os divisores **1** e **17**, portanto **17** é um número primo.
- 3) **10** tem os divisores **1, 2, 5** e **10**, portanto **10 não** é um número primo.

# Primos

- Um número é dito primo quando é apenas divisível por 1 e ele mesmo

```
int ehPrimo(int n){
    int qtdDivisores = 0;
    int i;
    for(i = 1 ; i <= n; i++){
        if (n%i == 0)
            qtdDivisores++;
    }

    if(qtdDivisores == 2 || n==1){
        return 1;
    }
    return 0;
}
```

# Fibonacci

Sequência: (1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, ...).

Essa sequência tem uma lei de formação simples: cada elemento, a partir do terceiro, é obtido somando-se os dois anteriores.

Veja:  $1+1=2$ ,

$2+1=3$ ,

$3+2=5$  e assim por diante.

# Fibonacci

```
int fibonacci(int num){  
    if(num==1 || num==2)  
        return 1;  
  
    else  
        return fibonacci(num-1) + fibonacci(num-2);  
}
```



Dúvidas???