

Combinações de Agrupamentos

Mário Leite

...

De acordo com a literatura matemática os três tipos de agrupamentos simples tratados na Análise Combinatória podem ser assim resumidos:

Arranjo Simples de n elementos dispostos p a p , com $p \leq n$, é uma escolha de p entre os n elementos na qual a ordem importa. Deve ser empregado, por exemplo, para determinar o pódio (1º, 2º e 3º lugares) de uma competição entre três jogadores; neste caso importa se é Plínio-Pedro-Paulo, ou Plínio-Paulo-Pedro, ou Paulo-Plínio-Pedro, ou Pedro-Plínio-Paulo, ou Pedro-Paulo-Plínio ou Paulo-Pedro-Plínio.

Permutação de n elementos distintos é um agrupamento ordenado de elementos; apenas isto! O número total de permutações pode ser calculado pela fórmula do fatorial dos n elementos, e deve ser empregada quando houver necessidade de saber quantas possibilidades existem de se organizar um número de elementos de forma distinta; por exemplo, o número de anagramas da palavra **PATO** é 24, calculado pelo produto: $4 \times 3 \times 2 \times 1$.

Combinações de n elementos tomados p a p são escolhas não ordenadas destes elementos, e dependem dos arranjos e das permutações. Os agrupamentos com elementos distintos não se alteram mudando-se apenas a ordem de posicionamento dos elementos no grupo, ao contrário do que ocorre com os arranjos. Deste modo com as três letras ABC só dá três combinações distintas com grupos de duas letras: AB, AC, e BC e não seis grupos se fosse o caso de Arranjo Simples.

As “Combinações Simples” são arranjos que se diferenciam apenas pela natureza de seus elementos; a ordem dos elementos não importa. Deste modo, se existe um conjunto X formado por n elementos tomados p a p , qualquer subconjunto de X formado por p elementos será uma combinação simples cujo total de agrupamentos é dado de por:

$$C_{n,p} = \frac{A_{n,p}}{P_p} = \frac{n!}{p!(n-p)!}$$

O programa “**Combinacoes**” implementa uma solução para determinar o número de combinações possíveis de n elementos tomados p a p , em três codificações diferentes: *Visualg*, *Small Basic* e *Python*.

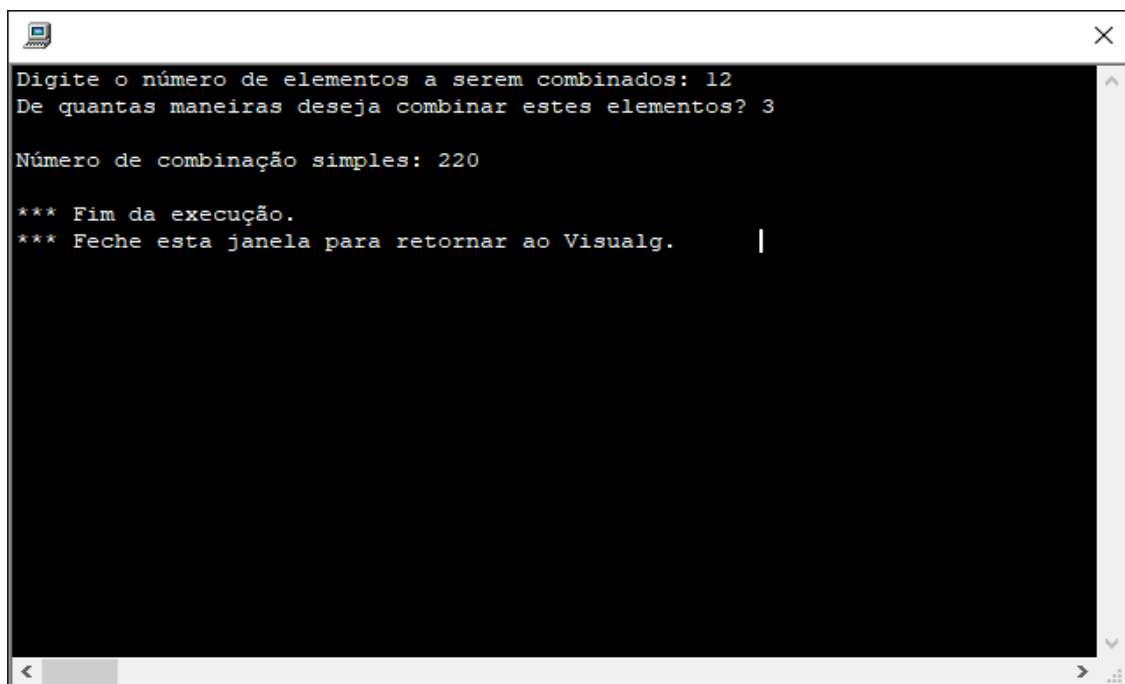
E conforme é colocado no programa, **Fat1** calcula os arranjos simples de n a p e **Fat2** dá o fatorial de p . Outro detalhe a ser observado no programa é quanto às declarações de **Fat1** e **Fat2** como **real** (em *Visualg*), embora o fatorial seja sempre inteiro; isto foi necessário devido ao fato de o fatorial crescer muito rapidamente, o que poderia resultar em estouro de memória (*overflow*) devido ao limite dos inteiros. Por outro lado, em *Small Basic* e em *Python* isto não é problema, pois estas duas linguagens não exigem declarações de variáveis, com suas bibliotecas matemáticas suportando números extremamente grandes. As figuras 1, 2 e 3 mostram os códigos-fonte do programa em *Visualg*, em *Small Basic*, e em *Python*, respectivamente.

VisuAlg 2.5 - D:\Livros\Livro7\Pastas para download\Arquivos para downloads dos leitores\Soluções dos Exemplos Exercícios Propostos\Códigos

Arquivo Editar Exibir Algoritmo Código Ferramentas Ajuda

```
Algoritmo "Combinacoes"
//Calcula o número de agrupamentos através de "Combinação Simples"
//Em Visualg
//Autor : Mário Leite
//E-mail : marleite@gmail.com
//-----
Var Fat1, Fat2: real
    Comb, j, n, p: inteiro
Inicio
    Escreva("Digite o número de elementos a serem combinados: ")
    Leia(n)
    Se (n<2) Entao
        n <- 2
    FimSe
    Escreva("De quantas maneiras deseja combinar estes elementos? ")
    Leia(p)
    Se (p<2) Entao
        p <- 2
    FimSe
    Escreval("") //salta linha
    Fat1 <- 1
    Para j De 1 Ate p Faca //calcula An,p
        Fat1 <- Fat1*(n-j+1)
    FimPara
    Fat2 <- 1
    Para j De 1 Ate p Faca //Calcula fatorial de p
        Fat2 <- Fat2*j
    FimPara
    Comb <- Int(Fat1/Fat2) //Calcula Cn,p
    Escreval("Número de combinação simples:", Comb)
FimAlgoritmo
```

Figura 1 - Código-fonte em Visualg



```
Digite o número de elementos a serem combinados: 12
De quantas maneiras deseja combinar estes elementos? 3

Número de combinação simples: 220

*** Fim da execução.
*** Feche esta janela para retornar ao Visualg.
```

Figura 1.1 - Saída do programa em Visualg

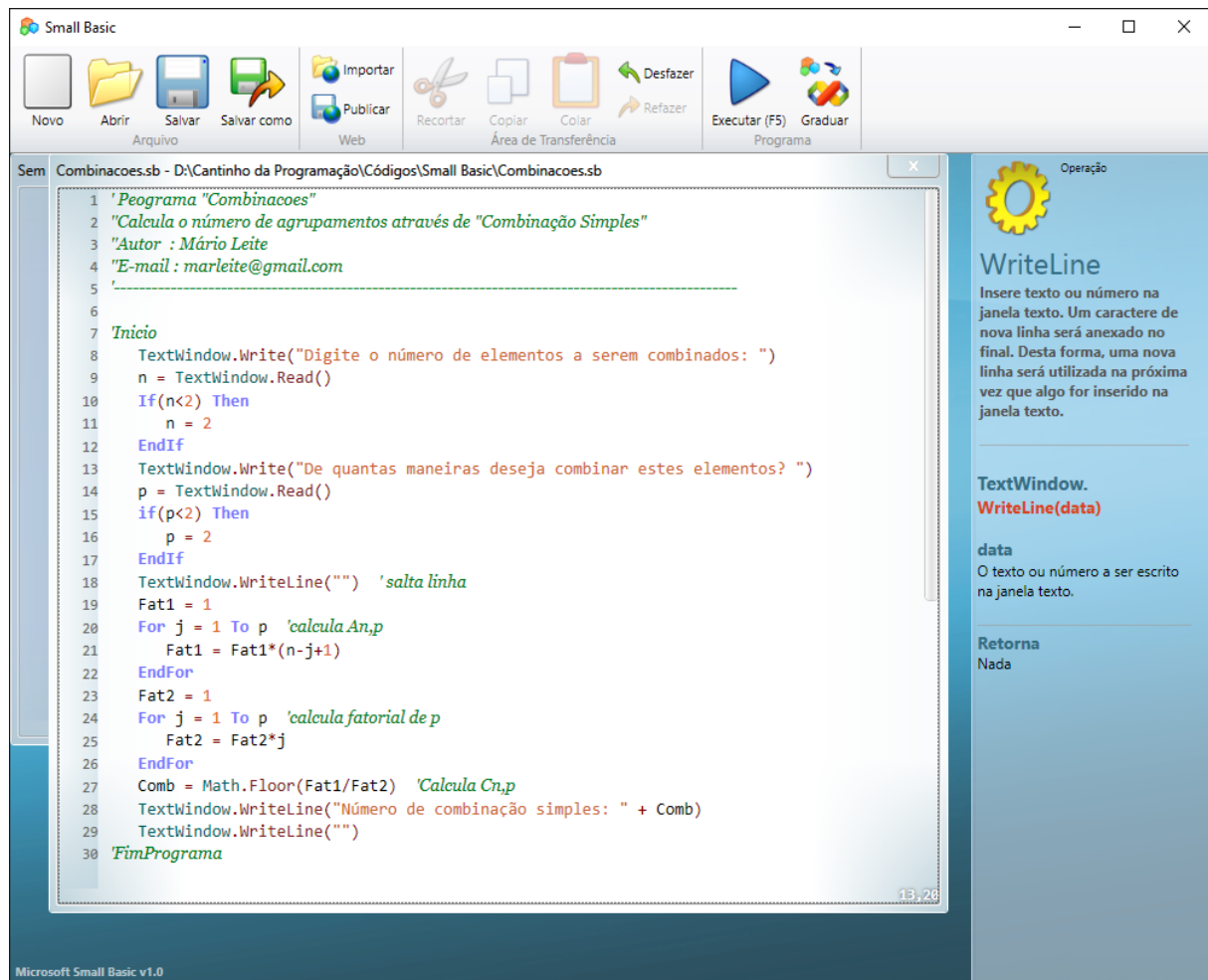


Figura 2 - Código-fonte em Small Basic

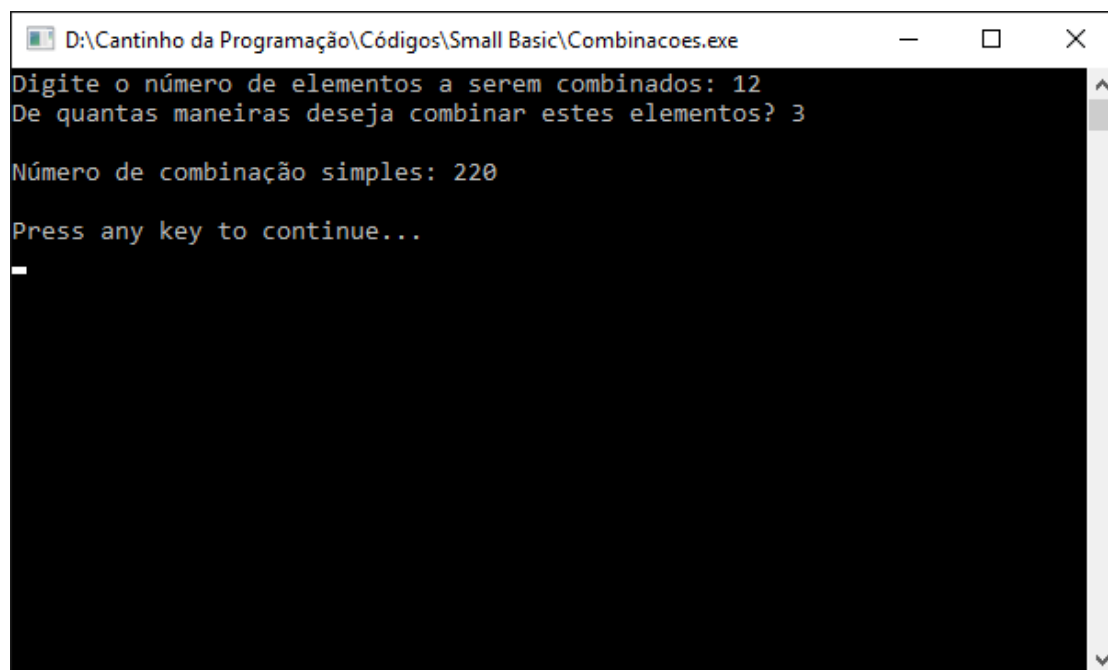


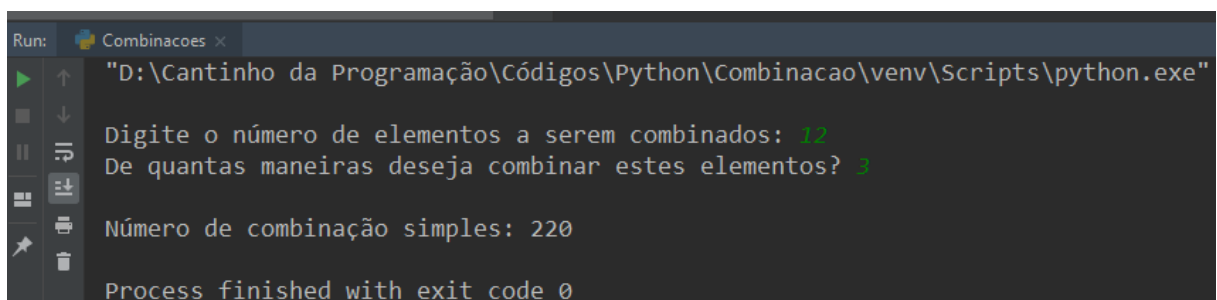
Figura 2.1 - Saída do programa em Small Basic

```
'''
Programa "Combinacoes"
//Calcula o número de agrupamentos através de "Combinação Simples"
//Em Python
//Autor : Mário Leite
//E-mail : marleite@gmail.com
//-----
'''

#Inicio
endif = "endif"
endfor = "endfor"

print("") #salta linha
n = int(input("Digite o número de elementos a serem combinados: "))
if(n<2):
    n = 2
endif
p = int(input("De quantas maneiras deseja combinar estes elementos? "))
if(p<2):
    p = 2
endif
print("") #salta linha
Fat1 = 1
for j in range(1, (p+1)): #calcula An,p
    Fat1 = Fat1*(n-j+1)
endfor
Fat2 = 1
for j in range(1, (p+1)): #fatorial de p
    Fat2 = Fat2*j
endfor
Comb = int(Fat1/Fat2) #Calcula Cn,p
print("Número de combinação simples:", Comb)
#FimAlgoritmo-----
```

Figura 3 - Código-fonte em Python



```
Run: Combinacoes x
"D:\Cantinho da Programação\Códigos\Python\Combinacao\venv\Scripts\python.exe"

Digite o número de elementos a serem combinados: 12
De quantas maneiras deseja combinar estes elementos? 3

Número de combinação simples: 220

Process finished with exit code 0
```

Figura 3,1 - Saída do programa em Python