

# IDL Tutorial



**AULA 09**

**SUBROTINAS  
ENTRADAS  
SAÍDAS  
FUNÇÕES  
SCRIPTS**

# SubRotinas



- O que são?
  - São os pequenos algoritmos que temos montados até agora.
- Declaração:
  - PRO nome
- Entradas/Saídas
  - PRO nome, in1, out1, in2, in3, inout1

# Entrada e Saídas



- Pode ser a mesma variável
- O nome não importa
- A posição na declaração é que importa
- Subrotinas podem ser chamadas por outras subrotinas ou pela linha de comando.

# Exemplo (arquivo simu.pro)



```
PRO func, in, out
```

```
    out = cos(in)
```

```
END
```

```
PRO simu
```

```
    x = dindgen(100)
```

```
    func, x, y
```

```
    PLOT, x, y
```

```
END
```

# Funções



- São comandos que executam determinado cálculo.
- São salvos em arquivos .PRO assim como as subrotinas. E possuem o mesmo nome de sua declaração.
- Declaração:
  - FUNCTION func, in

# Funções



- Saída?
  - Única, será o valor da função.
- Nome das entradas não importa.
- Entradas definidas pela ordem que aparecem.

# Exemplo



```
FUNCTION func, in  
    return, cos(in)  
END
```

```
PRO simu  
    x = dindgen(100)  
    y = func(x)  
    PLOT, x, y  
END
```

# Recursividade



- É quando uma função chama ela mesma.
- Utilidade:
  - Evitar loops
  - Realizar funções, cuja escrita estruturada torna-se-ia muito grande e/ou limitada, de uma forma a evitar as limitações e complexidade.



# Atividade



- 1) Criar uma função que calcule o fatorial de um número.
- 2) Criar uma função que calcule o produtório de uma função qualquer. Ex:
  - $F(x) = x^2$

# Torre de Hanoi



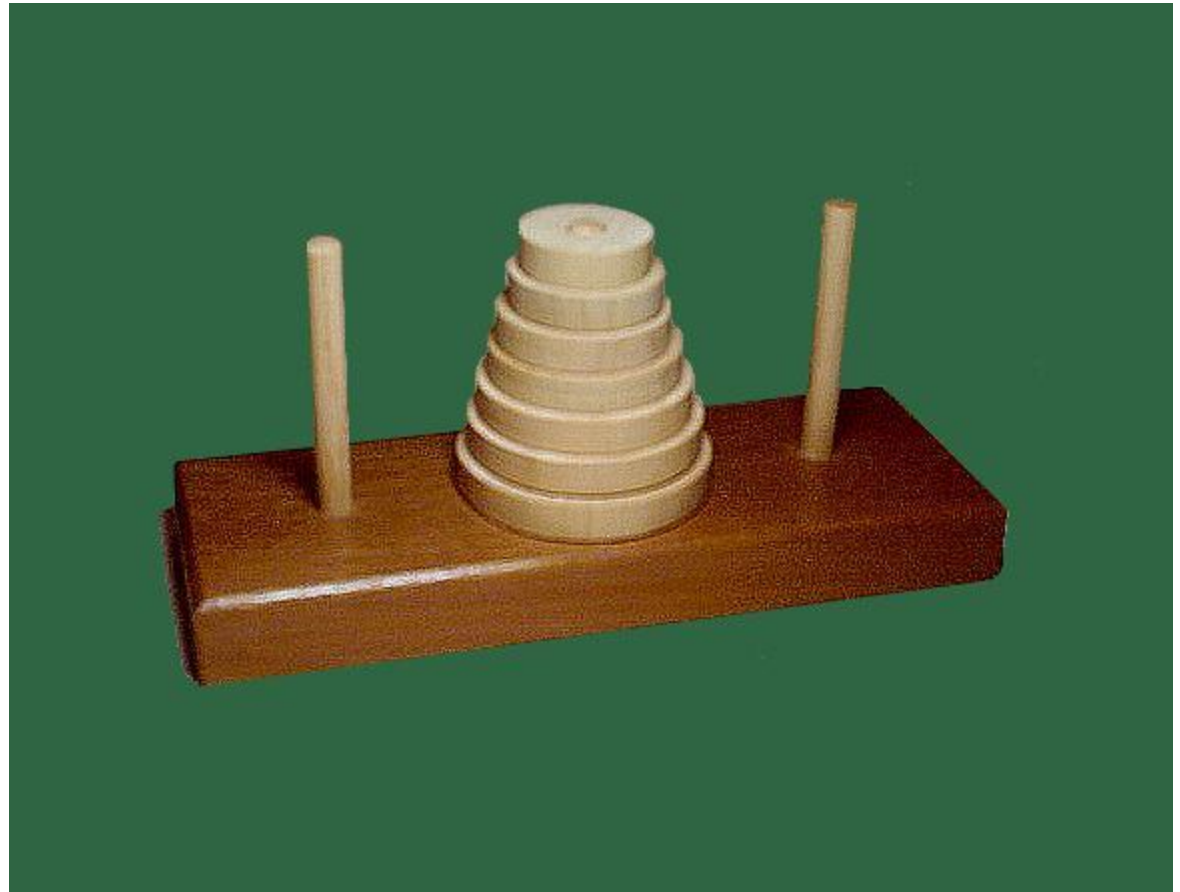
- Quebra-cabeças
- Principio:
  - 3 Torres
  - Diversos discos de diferentes tamanhos
- Regras
  - Move-se um disco por vez. Sempre o disco superior.
  - Um disco maior nunca pode ser posto em cima de um disco menor.



# Torre de Hanoi

Objetivo:

Transferir todos os discos de uma torre para outra.



# Atividade ESPECIAL



- Gerar um algoritmo capaz de:
  - Informar o número mínimo de jogadas capaz de alcançar o objetivo do jogo com determinado número de discos.
  - Informar quais as jogadas a serem feitas para afim de atingir o objetivo com o mínimo de jogadas possível.

# Conjunto de Funções



- É um arquivo onde diversas funções estão declaradas
- Para utilizar as funções é necessário compilar o arquivo.
- Compilando o conjunto de funções:
  - `.compile arquivo.pro`
- Só funciona em scripts e na linha de comando.

# Scripts



- Bloco de comandos funcionais.
- Particionamento de código-fonte.
- Scripts e variáveis globais.
- Usos: Particionamento de etapas, acoplamento de subrotinas, controle e liberdade da aplicação.

# Importância da Estruturação



- Estruturação
  - Subrotinas
  - Funções
  - Scripts
  - Projetos
- Todas são importantes.
- O uso eficiente de cada uma e a interação de todas elas é que tornam um simples algoritmo num poderoso projeto.

# Verdades sobre os grandes programas



- Grandes aplicações são feitas de pequenas partes.
- São estruturados de forma que suas partes possam ser feitas por pessoas diferentes.
- São planejadas como cada parte irá interagir com as demais.
- Rotina-mãe faz a interação entre os blocos funcionais (Subrotinas-filhas).



# Vantagens da estruturação



- Reduz o tamanho do código-fonte.
- Reduz a memória necessária para armazenar o algoritmo.
- Acelera a execução devido à não-necessidade de recarregamento de uma função na memória.
- Facilita a vetorização e a paralelização.

# Dúvidas?



**[HTTP://IDLTUTORIAL.BLOGSPOT.COM](http://idltutorial.blogspot.com)**

**[ANTONIOPAULOVP@GMAIL.COM](mailto:ANTONIOPAULOVP@GMAIL.COM)**

**[LUCIOMARASSI@GMAIL.COM](mailto:LUCIOMARASSI@GMAIL.COM)**