

Jogo de advinhação

Parte 1

Descrição

Computador “pensará” em um número e você tentará adivinhar

O computador apenas dirá se o número que você digitou é maior ou menor que o número a ser descoberto

Técnicas de programação: variáveis, estruturas condicionais e de repetição, números aleatórios

Será possível escolher o nível de dificuldade do jogo

Ao final de uma partida, o computador dirá quantos pontos você fez

O que você aprenderá

Ler entrada do teclado

Imprimir caracteres na tela

Armazenar e manipular valores na memória do computador (variáveis)

Executar operações matemáticas

Entender os diferentes tipos de variáveis

Tomar decisões a partir das entradas do teclado

Utilizar repetições para executar um trecho de código repetidas vezes

Primeiro passo

Abra uma IDE (utilizaremos o Geany, que é bem simples e prático)

No menu principal acesse Arquivo/Novo (com template)/main.c

A estrutura básica do programa estará pronta

Salve o arquivo com o nome adivinhacao.c (este será nosso código fonte)

Compile-o, construa o arquivo executável e execute-o, conforme os botões abaixo, respectivamente



Vá até o diretório onde o programa está armazenado e analize os arquivos criados



Dando boas vindas ao usuário

Até aqui nosso programa não faz nada, mas vamos fazer ele mostrar uma msg na tela

Para isso usaremos uma função chamada printf

```
printf("Bem-vindo ao Jogo de Adivinhação");
```

Sempre termine uma linha com ponto e vírgula

Execute o programa e veja o que acontece

Atividades

```
*****  
Bem-vindo ao Jogo de Advinhação  
*****  
:)
```

- 1 - Tente alterar o programa para a mensagem ficar conforme a figura acima
- 2 - Faça uma pesquisa e descubra a relação de printf com a linha #include <stdio.h>
- 3 - O que faz a instrução main? E a instrução return (ela pode ser comentada)?
- 4 - Por que se usa chaves { }?
- 5 - Diferencie código fonte de código de máquina. Como é feita esta conversão?

Declarando Variáveis

Vamos fazer o programa definir o “número secreto”

Esse número terá que ser armazenado na memória

O espaço de memória utilizado em um programa é o que chamamos de variável

Toda variável precisa de um nome

- O nome não pode conter espaços e caracteres especiais

- O C é case sensitive, ou seja, ele diferencia letras maiúsculas de minúsculas

Além do nome, precisamos definir o tipo da variável (caracter, texto, valor lógico ou número (inteiro, real, ...))

Declarando Variáveis

Vamos chamar nossa variável de numeroSecreto

Separar nomes compostos com letras maiúsculas ou _ (numero_secreto) é uma boa prática de programação

Ela será do tipo inteiro

Então a declaração dela será: `int numeroSecreto;`

Declarar todas as variáveis no início do código é uma boa prática de programação

Defina um valor desta variável: `numeroSecreto=30` (por exemplo)

Imprimindo o valor da variável

Para imprimir o valor de uma variável, temos que informar a função printf que uma variável vai ser impressa junto a mensagem

```
printf("\n O número secreto é %d \n", numeroSecreto);
```

Nesta linha de código %d informa que será impressa uma variável do tipo inteiro

Em seguida é informado o nome da variável

Esta linha de código não constará no nosso programa final, pois não queremos revelar o número secreto :)

Pesquise como outros tipos podem ser impressos

Lendo do teclado

Crie uma variável para armazenar o palpite do usuário do jogo

Usaremos a função scanf para ler do teclado o palpite do usuário

```
printf("\n Digite um número:");  
scanf("%d",&palpite);  
printf("\n O número digitado foi %d \n", palpite);
```

O uso do caractere & na função scanf será melhor explicado quando estudarmos ponteiros

Atividade

- 1 - A linguagem C é multiplataforma? O que seria necessário para executar este programa no Linux, Windows e MacOS?
- 2 - Pesquise o termo sintaxe de linguagem de programação.
- 3 - O que é um analisador léxico?
- 4 - O que é um analisador sintático?
- 5 - Como ocorre a análise léxica e sintática no processo de compilação?
- 6 - Complemente sua pesquisa com a análise semântica

Sugestão de leitura:

<https://pt.stackoverflow.com/questions/105438/qual-%C3%A9-a-diferen%C3%A7a-entre-erro-sint%C3%A1tico-e-sem%C3%A2ntico>

Atividade

7 - Digite no seu programa a linha: `integer numerosegreto §;`

Serão gerados dois erros. Identifique o erro léxico e o erro sintático.

8 - As linhas abaixo gerarão um erro léxico, sintático ou semântico?

```
int numeroSegreto, palpite;
```

```
int palpite;
```

Analogia com linguagem humana

Erro léxico: “Eu adolo sorvete”

Erro sintático: “Eu loja sorvete”

Erro semântico: “Eu durmo sorvete”