

Para os exercícios desta lista, use o https://www.onlinegdb.com/online_c_compiler . Deixe para compilar e escrever as coisas na mão mais pra frente, e com o suporte de software e compiladores adequados, no Sistema Operacional. Entre no link e nem olhe para o programa, apague tudo, não vale espiar. E de novo, faça TODOS os exercícios antes de olhar as referências.

Conceitos

Para começar precisamos de um arquivo de texto, com extensão ".c". Por exemplo, "teste.c". Dentro do arquivo, definimos o código do programa. O esqueleto de um programa em C pode ser definido como:

```
int main( ){  
    ...  
}
```

Observe a estrutura básica. Dentro dela, onde estão os "...", podemos colocar o que bem desejarmos. Todas as linhas que não forem estruturas, devem terminar com ponto e vírgula.

```
int main( ){  
    int x;  
    x = 1;  
    int y = 2;  
}
```

Podemos escrever, por exemplo, um programa em que temos as variáveis inteiras (por isso o "int") x e y, e elas recebem o valor 1 e 2, respectivamente.

Veja que tanto faz, declarar ela primeiro e depois atribuir um valor, ou já declarar e atribuir.

Temos aquilo que são conhecidos como tipos primitivos, que são tipos de dados que geram todos os outros. São eles:

- int : Números inteiros.
 - Exemplo: **2, 20233, 9, 0, 1, -12, -2034, ...**
- char: Letras e símbolos (Juntando os dois em um, são chamados Caracteres).
 - Repare que para mostrar para C que algo é um caracter, devemos colocá-lo entre aspas simples.
 - Exemplo: **'a', 'v', 'e', '&', '#', '!', '\$', '/', ...**
- float: Números racionais (Com vírgula(Que aqui é ponto) e casas depois dela).
 - Exemplo: **2.12 , 2.91 , 3.1416 , 99.99 , 0.0 , 0.1 , ...**
- Outros → Veremos mais à frente. Inclusive palavras, que é um conceito que precisa de mais definição. Por enquanto se contente com os números e letras.

As operações simples são feitas entre variáveis. Por hora, admita que são **+, -, *, /**.

Outro conceito importante é o de função. Não é aquela função do ensino médio e da matemática; função aqui, é um trecho de código que você declara, e chama depois usando parênteses. Por exemplo, "funcaoDaora()". Veja que engraçado, mas a main é uma função também, que na verdade é chamada pelo sistema operacional quando você executa o

programa, antes de qualquer outra do código, e é por isso que começamos colocar coisas por ela. Por hora, não construiremos mais funções, só chamaremos as prontas. Em C temos o conceito de biblioteca, que é basicamente o código de alguém, que já programou algumas funções ou tipos de dados úteis para você.

```
#include <stdio.h>

int main( ){
    int a = 0;
    printf("%d",a);
}
```

Para acrescentar uma biblioteca, basta colocar antes da main, um hashtag(#) include e o nome da biblioteca entre < >.

Podemos assim, usar a função **printf** da biblioteca **stdio**, que imprime coisas na tela. Para usá-la, basta escrever printf() e dentro colocar alguns parâmetros. O primeiro é a chamada "máscara", que deixa pontos em que serão substituídos outros

valores(No caso, a parte que vem entre aspas). Seguimos separando todos os outros argumentos restantes por vírgulas, e serão eles que serão colocados na máscara.

→ "%d" indica que você irá imprimir um número inteiro; em seguida separe o argumento e coloque outro argumento, que é o número a ser impresso no lugar do %d.

Para os tipos que vimos até agora a função printf utiliza as seguintes "porcentagens":

- %d : Para int
- %c : Para char
- %f : Para float

Você pode colocar varios %algo dentro das aspas, e só ir separando os argumentos na função printf, que ela trocará na ordem certinha. Tanto faz se é uma variável que guarda um valor(Como na imagem acima), ou o próprio valor. Por exemplo:

printf("Oi. Tenho %d anos. Mas já tive %d.", 23, 2);

→ Imprime na tela : Oi. Tenho 23 anos. Mas já tive 2.

Exercícios - Cole os código em [Code](#)

- 1) Faça o famoso "Hello World" em C, mostrando-o na tela.
- 2) Faça um programa que tenha quatro variáveis, w,x,y,z. Elas recebem, **13, 15.5, a, 80**, respectivamente. Imprima elas de forma simples, separadas por espaços mesmo.
- 3) Faça um programa guarde 21 e 92 em variáveis, some as duas e mostre na tela o resultado.

Code

"cole aqui"

"cole aqui"

"cole aqui"

Referências

- https://www.youtube.com/watch?v=GiCt0Cwcp-U&list=PL8iN9FQ7_jt4DJbeQqv--jpTy-2gTA3Cp
- https://www.youtube.com/watch?v=q51cHsgRHU4&list=PL8iN9FQ7_jt4DJbeQqv--jpTy-2gTA3Cp&index=2
- https://www.youtube.com/watch?v=07YPObbEpU8&list=PL8iN9FQ7_jt4DJbeQqv--jpTy-2gTA3Cp&index=3