

Glossário dos Degraus - Fundamentos de Programação

Este glossário reúne as palavras novas de cada bloco da disciplina, na ordem em que elas aparecem nas aulas. Ele existe para um momento específico: quando alguma palavra travar o seu entendimento durante a aula ou o exercício. Procure a palavra aqui, leia a explicação curta e, se quiser ir além, o material completo do bloco está disponível no Teams desde a primeira aula. Sentir-se perdido com um termo novo é parte normal do primeiro semestre; ficar perdido é que não precisa.

Antes de programar - o ambiente e a rotina

Palavra	O que significa
Algoritmo	Sequência de passos para resolver um problema, como uma receita de bolo: se a ordem estiver errada, o resultado sai errado.
Lógica de programação	O jeito de pensar que transforma um problema em passos que o computador consegue seguir.
Fluxograma	O desenho do algoritmo, com caixas e ligações, feito antes do código para enxergar o caminho da solução.
Linguagem C99	A linguagem usada na disciplina: o idioma em que escrevemos as ordens para o computador.
Código-fonte	O texto do programa que você escreve. É o arquivo com extensão .c.
Dev-C++	O programa em que escrevemos, compilamos e executamos o código na disciplina (Embarcadero Dev-C++ 6.3).
Compilar	Traduzir o código-fonte para a língua da máquina. Se houver erro de escrita, a tradução para e o compilador aponta a linha do problema.
Executar (rodar)	Colocar o programa traduzido para funcionar e ver o resultado na tela.
Console	A tela preta onde o programa conversa com quem usa: mostra mensagens e recebe o que é digitado.
Erro de compilação	Aviso de que algo no texto do código impede a tradução. Ler a mensagem de erro, que quase sempre indica a linha, é parte do trabalho de programar.

Degrau 1 - Saída de dados, variáveis, operadores e entrada de dados

Palavra	O que significa
Biblioteca / #include	Caixa de ferramentas pronta que o programa pede emprestada no topo do código. A stdio.h traz o printf e o scanf.
int main(void)	O ponto de partida do programa: a execução sempre começa aqui.
Chaves { }	Marcam onde um bloco de código começa e onde termina.

Ponto e vírgula ;	Encerra cada instrução, como o ponto final encerra uma frase.
Comentário	Anotação para pessoas lerem; o computador ignora. Linha única com // ou bloco com /* e */.
printf	Escreve na tela: é a saída de dados do programa.
\n	Dentro do texto do printf, pula para a linha de baixo.
Variável	Espaço na memória com um nome, que guarda um valor que pode mudar. Pense numa gaveta etiquetada.
Tipo de dado	O que a gaveta aceita guardar: int para números inteiros, float para números com vírgula, char para um caractere.
Declarar	Criar a variável, informando o tipo e o nome: int idade;
Atribuir (=)	Guardar um valor na variável: idade = 20; Atenção: um único sinal de igual.
Operadores aritméticos	As contas: + soma, - subtração, * multiplicação, / divisão e % resto da divisão.
Especificador de formato	O marcador de lugar no texto do printf e do scanf: %i inteiro, %f real, %.2f real com duas casas, %c caractere, %s texto.
scanf	Lê o que o usuário digita: é a entrada de dados do programa.
& (e comercial)	No scanf, indica em qual variável o valor digitado será guardado.
Processamento	A parte do programa que transforma a entrada na saída: a conta, a comparação, a decisão.

Degrau 2 - Estrutura condicional e estrutura de seleção

Palavra	O que significa
Condição	Pergunta com resposta verdadeiro ou falso que decide o caminho do programa.
Operadores relacionais	As comparações: > maior, < menor, >= maior ou igual, <= menor ou igual, == igual, != diferente.
== e =	Dois iguais comparam; um igual guarda. Trocar um pelo outro é um dos erros mais comuns do semestre.
Operadores lógicos	Combinam condições: && (e), (ou), ! (não).
if / else if / else	Se, senão se, senão: os caminhos possíveis de uma decisão.
Bloco	O conjunto de instruções entre chaves que pertence a um caminho da decisão.
switch / case	Seleção por valor: compara uma variável com várias opções, uma por case. É a estrutura típica de menus.
break	Encerra o case atual do switch, para a execução não invadir o próximo.
default	O caso "nenhuma das anteriores" do switch: executa quando nenhum case combina.

Degrau 3 - Estrutura de repetição

Palavra	O que significa
Laço (repetição)	Trecho de código que se repete enquanto uma condição for verdadeira.
for	Laço com início, condição e passo na mesma linha. Bom quando se sabe quantas vezes repetir.
while	Laço que testa a condição antes de executar: pode não rodar nenhuma vez.
do while	Laço que executa primeiro e testa depois: roda pelo menos uma vez. Típico de menus.
Variável de controle	A variável que conta as voltas do laço: o i do for(int i = 0; i < n; i++).
Condição de parada	O que precisa acontecer para o laço terminar.
Laço infinito	Quando a condição nunca fica falsa e o programa repete para sempre. Quase sempre, faltou atualizar alguma variável dentro do laço.
Formas contraídas	Atalhos de escrita: x++ equivale a x = x + 1; x-- a x = x - 1; x += y a x = x + y; e assim com -=, *= e /=.

Degrau 4 - Vetores e matrizes (estruturas homogêneas)

Palavra	O que significa
Vetor	Uma fila de gavetas do mesmo tipo sob um único nome: float notas[30] cria trinta gavetas chamadas notas.
Índice	O número da gaveta. Começa em 0: a primeira posição é notas[0], e a última de trinta é notas[29].
Percorrer	Visitar as posições do vetor uma a uma, quase sempre com um for.
Matriz	Uma tabela de gavetas, com linhas e colunas. Precisa de dois índices: matriz[linha][coluna].
Estrutura homogênea	O nome geral de vetores e matrizes: estruturas em que tudo é do mesmo tipo.

Degrau 5 - Funções

Palavra	O que significa
Função	Bloco de código com nome próprio que executa uma tarefa. Escreve-se uma vez e usa-se quantas vezes for preciso.
Chamar uma função	Mandar a função executar, escrevendo o nome dela no código.
Parâmetro	O que a função recebe para trabalhar, declarado na definição da função.
Argumento	O valor que você envia para a função no momento da chamada.
Retorno (return)	O resultado que a função devolve para quem a chamou.
void	Indica que a função não devolve nada (ou

	que não recebe nada).
Protótipo	O anúncio da função antes do main, informando nome, parâmetros e retorno, para o compilador saber que ela existe.

Material de apoio da disciplina Fundamentos de Programação - Prof. Manfred Heil Junior. Uso em sala e no Teams.

Referências

BHARGAVA, Aditya Y. Entendendo algoritmos: um guia ilustrado para programadores e outros curiosos. São Paulo: Novatec, 2017.

LOPES, Anita; GARCIA, Guto. Introdução à programação: 500 algoritmos resolvidos. Rio de Janeiro: Elsevier, 2002.

MIZRAHI, Victorine Viviane. Treinamento em linguagem C. 2. ed. São Paulo: Pearson Prentice Hall, 2008.

HEIL JUNIOR, Manfred. Apostilas da disciplina Fundamentos de Programação, blocos 01 a 07. Jaraguá do Sul: material didático do professor, 2026.

HEIL JUNIOR, Manfred. Aprendendo a Programar com Prof. Manfred. Livro didático em elaboração, 2026.