



Universidade de Brasília
Faculdade UnB Gama
Disciplina: Estrutura de Dados e Algoritmos - EDA

Revisão de Linguagem C

Estruturas de Dados Compostas Homogêneas

Prof. Nilton Correia da Silva

23 de outubro de 2024

Estruturas de Dados Compostas Homogêneas

Definição

Vetores

Códigos

Matrizes 2D

Códigos

Matrizes 3D

Códigos

Exercícios

Matrizes 1D

Matrizes 2D

Matrizes 3D

Estruturas de Dados Compostas Homogêneas

Definição



- ▶ Estruturas de dados compostas homogêneas são tipos de estruturas de dados que consistem em conjuntos onde todos os elementos são do mesmo tipo.
- ▶ Manipular dados é relativamente simples porque todos os elementos são tratados da mesma maneira.
- ▶ São usadas em uma variedade de aplicações que requerem armazenamento e processamento de grandes quantidades de dados similares: Exemplos: Imagens, Vídeos, Dados Estatísticos.

Declaração e inicialização

```
int vetor[10] = 0; // Declara um vetor de inteiros com 10  
elementos, todos inicializados com 0.
```

Acesso e Modificação

```
vetor[5] = 15; // Atribui o valor 15 ao sexto elemento do  
vetor.
```

Iteração

```
1 for(int i = 0; i < 10; i++) {  
2     printf("%d ", vetor[i]);  
3 }
```

Busca

```
1  #include <stdio.h>
2  // Função para buscar um caractere no vetor
3  int Busca(char vet[], int tam, char target) {
4      for (int i = 0; i < tam; i++) {
5          if (vet[i] == target)
6              return i;
7      }
8  // Retorna -1 se o caractere não for encontrado
9      return -1;
10 }
```

Busca...

```
1  int main() {
2      char Letras[10] = \
3      {'A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J'};
4      char caracterParaBuscar;
5      int posicao;
6      printf("Digite um caractere para buscar no vetor: ");
7      scanf(" %c", &caracterParaBuscar);
8      posicao = Busca(Letras, 10, caracterParaBuscar);
9      if (posicao != -1)
10         printf("Encontrado na posição: %d\n", posicao);
11     else
12         printf("Não encontrado.\n");
13     return 0;
14 }
```

Declaração e inicialização

```
int matriz[3][4] = {{0}}; // Declara uma matriz 3x4 de  
inteiros, todos inicializados com 0.
```

Acesso e Modificação

```
matriz[2][1] = 15; // Atribui o valor 15 ao elemento na  
terceira linha e segunda coluna.
```


Iteração

```
1  for(int i = 0; i < 3; i++) {  
2      for(int j = 0; j < 4; j++) {  
3          printf("%d ", matriz[i][j]);  
4      }  
5      printf("\n"); // Nova linha para cada linha da matriz  
6  }
```

Leitura e Impressão

```
1  #include <stdio.h>
2  int main() {
3      float Notas[80][3], soma;
4      printf("Digite as 3 notas dos 80 alunos:\n");
5      for (int i = 0; i < 80; i++)
6          for (int j = 0; j < 3; j++)
7              scanf("%f", &Notas[i][j]);
8      for (int i = 0; i < 80; i++) {
9          soma = 0.0;
10         for (int j = 0; j < 3; j++)
11             soma += Notas[i][j];
12         printf("Media %d: %.2f\n", i + 1, soma / 3.0);
13     }
14     return 0;
15 }
```

Declaração e inicialização

```
int matriz[2][3][4] = {{{0}}}; // Declara uma matriz 2x3x4  
de inteiros, todos inicializados com 0.
```

Acesso e Modificação

```
matriz[1][2][3] = 15; // Atribui o valor 15 ao elemento na  
segunda camada, terceira linha e quarta coluna.
```

Iteração

```
1  for(int i = 0; i < 2; i++) {  
2      for(int j = 0; j < 3; j++) {  
3          for(int k = 0; k < 4; k++) {  
4              printf("%d ", matriz[i][j][k]);  
5          }  
6          printf("\n");  
7      }  
8      printf("\n");  
9  }
```

Leitura e Impressão

```
1      // Cálculo e impressão das médias das turmas
2      printf("\nMedias das Turmas:\n");
3      for (int t = 0; t < 5; t++) {
4          soma = 0.0;
5          for (int a = 0; a < 80; a++) {
6              for (int n = 0; n < 3; n++) {
7                  soma += Notas[t][a][n];
8              }
9          }
10         media = soma / (80 * 3);
11         printf("Media da Turma %d: %.2f\n", t + 1, media);
12     }
13     return 0;
14 }
```

Leitura e Impressão...

```
1  #include <stdio.h>
2
3  int main() {
4      // Matriz[Turmas][Alunos][Notas]
5      float Notas[5][80][3], media, soma;
6      for (int t = 0; t < 5; t++) {
7          printf("Digite as notas da turma %d:\n", t + 1);
8          for (int a = 0; a < 80; a++) {
9              printf("Aluno %d, Turma %d:\n", a + 1, t + 1);
10             for (int n = 0; n < 3; n++) {
11                 printf("Nota %d: ", n + 1);
12                 scanf("%f", &Notas[t][a][n]);
13             }
14         }
15     }
```

Descrição do Problema

Você precisa criar um sistema de controle de estoque para um supermercado que mantém a contagem de produtos diferentes em um vetor. Cada índice do vetor representa um tipo de produto (por exemplo, índice 0 para arroz, índice 1 para feijão, etc.), e o valor em cada índice indica a quantidade atual desse produto no estoque.

Objetivo

Escreva um programa que inicialize este vetor com quantidades aleatórias para 10 produtos diferentes, permita ao usuário “comprar” produtos (diminuindo a quantidade do estoque), e adicionar novos produtos ao estoque. O programa deve imprimir o estado atual do estoque após cada operação.

Descrição do Problema

Um teatro quer implementar um sistema de reservas online onde os assentos podem ser visualizados e selecionados através de uma matriz 2D. Cada linha representa uma fileira e cada coluna um assento naquela fileira, onde 0 representa um assento disponível e 1 um assento ocupado.

Objetivo

Desenvolva um programa que inicialize uma matriz para um teatro com 15 fileiras e 20 assentos por fileira, permita ao usuário escolher um assento, verifique a disponibilidade e atualize o status do assento de disponível para ocupado após a reserva. O programa deve ser capaz de mostrar a matriz de assentos atualizada após cada reserva.

Descrição do Problema

Imagine um edifício com vários andares, onde cada andar tem várias salas, e a temperatura de cada sala é monitorada. As temperaturas de todas as salas são armazenadas em uma matriz 3D, onde a primeira dimensão representa o andar, a segunda a sala nesse andar, e a terceira dimensão representa leituras de temperatura ao longo do dia.

Objetivo

Escreva um programa que inicialize aleatoriamente as temperaturas para um edifício de 5 andares, com 10 salas por andar e 24 leituras de temperatura por dia (uma por hora). O programa deve permitir ao usuário selecionar um andar e uma sala e exibir as temperaturas dessa sala ao longo do dia. Além disso, o programa deve calcular e exibir a temperatura média de cada andar.