

1. DCC – Departamento de Ciência da Computação
2. AGT – Algoritmos T/C
3. Profa Everlin Costa Marques

Conteúdo: Matrizes

Como ler uma matriz quadrada

|  |
| --- |
| #include <stdio.h>  #define L 3  int main(){  int mat[L][L];  int ilinha, icoluna;  for (ilinha=0; ilinha < L; ilinha++){  for(icoluna=0; icoluna<L; icoluna++){  printf("pos(%i,%i)=", ilinha, icoluna);  scanf("%i", &mat[ilinha][icoluna] );  }  }  printf("\nSaida\n");  for (ilinha=0; ilinha < L; ilinha++){  printf("\n");  for(icoluna=0; icoluna<L; icoluna++){  printf("%i\t", mat[ilinha][icoluna] );  }  }  return 0;  } |

Encontrar diagonal principal em matriz quadrada

|  |
| --- |
| printf("\nDiagonal principal\n");  for (ilinha=0; ilinha < L; ilinha++){  printf("\n");  for(icoluna=0; icoluna<L; icoluna++){  if ( ilinha==icoluna)  printf("%i\t", mat[ilinha][icoluna] );  }  } |

Encontrar diagonal secundária em matriz quadrada

|  |
| --- |
| printf("\nDiagonal secundaria\n");  for (ilinha=0; ilinha < L; ilinha++){  printf("\n");  for(icoluna=0; icoluna<L; icoluna++){  if ( ilinha + icoluna == L-1)  printf("%i\t", mat[ilinha][icoluna] );  }  } |

## Código 1 para alocar matriz dinamicamente

Você pode ler um valor para tamanho da matriz do usuário,

scanf("%i", &N);

mas use malloc ( incluir a biblioteca <stdlib.h> ) .

Esse código assume que uma matriz de inteiros será criada ( representada por int \*\*M)

----------------------------------------

#include <stdio.h>

#include<stdlib.h>

int main(void) {

int n,m ,lin, col;

int \*\*M; // M será sua matriz, mas usaremos um ponteiro ( como vetor de vetor )

printf("\nDigite a ordem da matriz:\n");

scanf("%i", &m);

M = malloc (m \* sizeof (int \*));

for (int i = 0; i < m; ++i)

M[i] = malloc (n \* sizeof (int));

for (lin=0; lin<m;lin++){

for(col=0; col<m; col++){

printf("\n(%i,%i)",lin, col);

scanf("%i", &M[lin][col]);

}

}

printf("\nsaida\n");

for (lin=0; lin<m;lin++){

printf("\n");

for(col=0; col<m; col++){

printf("%i\t",M[lin][col]);

}

}

free(M);

return 0;

}

Ler vários nomes em uma “lista” ( matriz de strings)

|  |
| --- |
| #include <stdio.h>  #include<string.h>  #define L 4  #define TAMSTR 250  int main(){  char alunos[N][200]={"Felipe M", "Gustavo S", "Lucas A", "Andre Trent", "Elija", "Gustavo Porto", "Vinicius Souza", "Vinicius Dias","Rafael Souza", "Thiago Rafael", "Walter" };  int lin, col;  /\* ou entre os nome via digitação no teclado  for(lin=0 ; lin < L; lin++){  printf("\nDigite o nome (%i):", lin);  fgets(alunos[lin], 200, stdin);  }\*/  printf("\nSaida\n");  for(lin=0 ; lin < L ; lin++){  printf("\n");  printf("Estudante %i :%s ", lin , alunos[lin]);  }  return 0;  } |

#### Refer***ências bibliográficas***

Fontes: [*https://www.ime.usp.br/~pf/algoritmos/aulas/aloca.html*](https://www.ime.usp.br/~pf/algoritmos/aulas/aloca.html)

[*https://www.inf.ufpr.br/roberto/ci067/14\_alocmat.html*](https://www.inf.ufpr.br/roberto/ci067/14_alocmat.html)