

# Agenda de Casas

# Cadastro

- Ao iniciar, o programa perguntará ao operador “quantas casas serão cadastradas?”, então alocará dinamicamente um vetor com a quantia de casas informada.
  - O programa apresentará as seguintes opções em um menu:
    - [1] Cadastrar Casa – onde será feito o cadastro de uma casa. Se o operador tentar cadastrar mais casas do que o número informado inicialmente o programa emitirá um aviso;
    - [2] Cadastrar Morador – onde será feito o cadastro de uma pessoa dentro de uma casa. Inicialmente o operador deverá informar o código da casa. Se este código for inexistente na lista de casas cadastradas, o programa emitirá um aviso. Só cabem 4 pessoas em cada casa, esse número não deve ser ultrapassado, caso o operador tente inserir 5 pessoas, será emitido um aviso e o programa retorna ao menu inicial.
    - [3] Listas casas – onde serão listadas as informações das casas e de seus moradores.
    - [4] Apagar um morador – onde o morador de uma determinada casa será excluído, tendo como critérios os códigos da casa e do morador.
    - [5] Apagar Casa - onde uma casa com todos seus moradores será excluída, tendo como critério o código da casa.

# Estruturas

```
typedef struct{  
    int CodigoPessoa;  
    char Nome[20];  
    int Idade;  
} Pessoa;
```

```
typedef struct{  
    int CodigoCasa;  
    char Descricao[15];  
    Pessoa lista_Pessoa[4];  
} Casa;
```

# Alocação dinâmica de memória

- Os comandos abaixo alocam dinamicamente um inteiro e depois o liberam:

```
#include <stdlib.h>
int *pi;
pi = (int *) malloc (sizeof(int));
...
free(pi);
```

- A função malloc não tem um tipo específico. Assim, (int \*) converte seu valor em ponteiro para inteiro. Como não sabemos necessariamente o comprimento de um inteiro (2 ou 4 bytes dependendo do compilador), usamos como parâmetro a função sizeof(int).

# Menu com Switch

```
int op;
printf ("Digite uma opcao [0-5]: ");
scanf ("%d",&op);
switch (op)
{
    case 0:
        printf ("\n\nSair ... \n");
        break;
    case 1:
        printf ("\n\nCadastro Casa.\n");
        break;
    case 2:
        printf ("\n\nCadastro Morador.\n");
        break;
    case 3:
        printf ("\n\nLista Casas.\n");
        Break;
    case 4:
        printf ("\n\nApagar Morador\n");
        break;
    case 5:
        printf ("\n\nApagar Casa.\n");
        break;
    default:
        printf ("\n\nO numero nao esta entre as alternativas\n");
}
```