

Exercícios 05 – TDA

5.1 Defina o tipo **Cheque** para armazenar o número, o valor e a situação (0-Em branco, 1-Emitido, 2-Compensado) de um cheque.

Implemente as seguintes operações:

Inclua as definições e protótipos no arquivo **Cheque.h**

A implementação das funções no arquivo **Cheque.c**

criaCheque Saída: um cheque Entrada: Número do cheque Descrição: Atribui os valores iniciais. O valor e a situação do cheque deve ser zeradas.	exibeCheque Entrada: Um cheque Descrição: Exibe um uma única linha as 3 informações contidas no cheque.
obtemSituacao Entrada: um cheque Retorno: O código da situação do cheque	obtemValor Entrada: um cheque Retorno: O valor do cheque
obtemNumero Entrada: um cheque Retorno: O número do cheque	compensaCheque Entrada/Saída: um cheque Retorno: Código 0 (sucesso), 1 (erro) Descrição: Altera a situação do cheque para 2 (compensado). O cheque só pode ser compensado se a situação é igual a 1 (emitido)
emiteCheque Entrada/Saída: um cheque Entrada: Valor do cheque Retorno: Código 0 (sucesso), 1 (erro) Descrição: Atribui um valor ao cheque e altera a situação para o código 1 (Emitido). O cheque só pode ser compensado se a situação é igual a 0 (em branco)	

Escreva um programa para ler dois números e criar dois cheques armazenando-os em 2 variáveis (usar **criaCheque**).

Ler 2 valores e emitir os 2 cheques (usar **emiteCheque**).

Escrever na tela os dados dos 2 cheques (usar **exibeCheque**)

Compensar o que possui maior valor (usar **obtemValor** para obter o valor de cada cheque e **compensaCheque** para fazer a compensação).

Escreva somente a código da situação dos 2 cheques (usar **obtemSituacao**).

Escreva somente a número dos 2 cheques (usar **obtemNumero**).

5.2 Defina o tipo **Talao** para armazenar o número da conta, e um vetor com 10 cheques.

Implemente as seguintes operações:

Inclua as definições e protótipos no arquivo **Talao.h**

A implementação das funções no arquivo **Talao.c**

criaTalao Saída: um talão Entrada: Número da conta, número do primeiro cheque do talão. Descrição: Inicializa os 10 cheques do talão. A numeração deve começar no número do primeiro passado como parâmetro. (usar criaCheque)	exibeTalao Entrada: um talão Descrição: Exibe o número da conta e todos os cheques do talão (usar exibeCheque).
--	--

emiteChequeDoTalao Entrada/Saída: um talão Entrada: Número do cheque, valor do cheque Retorno: 0 (Sucesso) 1 (caso não seja possível emitir o cheque). Descrição: Altera a situação do cheque cujo número é passado como parâmetro para o código 1 (emitido). (usar obtemNumero , emiteCheque)	CompensaChequeDoTalao Entrada/Saída: um talão Entrada: Número do cheque, valor do cheque Retorno: 0 (Sucesso) 1 (caso não seja possível compensar o cheque). Descrição: Altera a situação do cheque cujo número é passado como parâmetro para o código 2 (compensado). (usar obtemNumero , compensaCheque)
somaChequesDoTalao Entrada: um talão, situação dos cheques a ser somado (0 em branco, 1 emitido, 2 compensado) Retorno: a soma dos valores dos cheques cuja situação foi passa como parâmetro. (usar obtemSituacao , obtemValor)	

Escreva um programa para ler o número da conta, o número do primeiro cheque do talão e criar um talão (usar **criaTalao**).

A seguir escrever o seguinte cardápio e ler o código da operação a ser executada.

```

0.Fim
1.Exibe talao
2.Emite cheque
3.Compensa cheque
4.Soma cheques do talao

```

Executar a operação escolhida conforme a descrição abaixo:

- 1:** Exibir o talão de cheques (usar **exibeTalao**)
- 2:** Ler o número e o valor de um cheque e executar a sua emissão (usar **emiteChequeDoTalao**). Imprimir a palavra “SUCESSO” se a operação foi executada com sucesso ou “ERRO” em caso de erro.
- 3:** Ler o número de um cheque e executar a sua emissão (usar **compensaChequeDoTalao**). Imprimir a palavra “SUCESSO” se a operação foi executada com sucesso ou “ERRO” em caso de erro.
- 4:** Ler o código de uma situação e exibir a soma dos valores dos cheques que possuem a situação informada (usar **somaChequesDoTalao**).