

USP – ICMC – SSC

**SSC0501 - Introdução à Ciência da Computação I (Teórica)**

**Professor responsável:** *Fernando Santos Osório*  
**Semestre:** 2015/1  
**Horário:** Ter. 21h-22h40 e Sexta 19h-20h40

**E-mail:** fosorio @ icmc.usp.br  
fosorio @ gmail.com  
**Web:** <http://www.icmc.usp.br/~fosorio/>

**Nome do Aluno:** \_\_\_\_\_

**Número USP :** \_\_\_\_\_

**DATA: 28 / 04 / 2015**

***PROVA TEÓRICA – P1 (Prova Escrita)***

1) Faça os “testes de mesa” para determinar a saída dos programas abaixo.

**1.1. [1.0 pt]**

```
#include<stdio.h>
int main() {
    int vet[6] = {14,0,10,12,35,11};
    int A, B, C, D;

    D = 4;
    A = vet[0];
    B = vet[2];
    C = vet[4];
    printf("Antes:\n A:%d\n B:%d\n C:%d\n D:%d\n",A,B,C,D);
    while ( (A < C) && (B > 0) ) {
        A=A+1;
        B=B-1;
        C=vet[D];
    }
    D++;
    printf("Depois:\n A:%d\n B:%d\n C:%d\n D:%d\n",A,B,C,D);
    return 0;
}
```

Resposta:

Antes do While

vet[1] =

A =

B =

C =

D =

Depois do While

vet[1] =

A =

B =

C =

D =

**1.2. [1.0 pt]**

```
#include<stdio.h>
int main() {
    int vet[10] = {3,6,1,5,2,4,7,-1,-1,-1};
    int i = 0; int C=0; int T=0;

    while (i >= 0) {
        T = T + vet[C];
        printf("C:%d = %d \n", C++, vet[i]);
        i = vet[i];
    }
    printf("T=%d\n",T);
    return 0;
}
```

Resposta:

C:0 =

C:1 =

C:2 =

C:3 =

C:4 =

C:5 =

C:6 =

C:7 =

C:8 =

C:9 =

T =

2) **Implemente o seguinte programa de controle de contas bancárias (“caixa de banco”), sendo que o programa tem 3 etapas a serem implementadas:**

1. **Início de atividades do dia: informar as contas e os saldos de cada conta**
2. **Atendimento de clientes: Saques/Pagamentos (saída) e Depósitos (entrada)**
3. **Consolidação dos dados ao final do dia, listando as contas e o saldo atual**

**Etapa 1.** Início de atividades do dia: informar as contas e os saldos de cada conta

Entrada de Dados: Faça a primeira parte do programa para ler os dados das contas bancárias e seus saldos. As contas bancárias são definidas por 2 informações: **código da conta** (valor numérico de até 5 dígitos que representa o número da conta no banco de um determinado cliente) e **saldo atual da conta** (valor em dinheiro que está disponível na conta, se o saldo for positivo, ou, valor devido pelo cliente, se o saldo for negativo). Estes dados devem ser armazenados em memória (vetor(es)), sendo que o banco tem um limite máximo de até 1000 clientes (contas distintas) que podem ter gerenciadas nesta agência. A entrada de dados ocorre da seguinte forma: primeiro o usuário informa a data atual (texto que não precisa ser validado), depois o usuário deve indicar **o código da conta**, e depois é **lido o valor do saldo da conta** em Reais (positivo ou negativo). Após, será perguntado ao usuário se ele deseja entrar mais dados (0:Não, 1:Sim), se ele responder com 1 então o programa continuará lendo mais dados de outras contas, caso ele responda com 0, o programa irá passar para a próxima etapa. Esta resposta deve ser validada, onde só será aceita a resposta 0 ou 1, qualquer outro valor o usuário deverá digitar novamente [3.0 pontos].

Exemplo da tela de entrada inicial de dados do programa: (procure seguir o exemplo!)

```
<< BANCO ACME >>
>> Entrada de Dados Inicial <<
Data atual (dd/mm/aa): 28/04/15

Nro. da Conta: 12345
Saldo da Conta: 100.00
Deseja entra mais dados (0:Não, 1:Sim) ? 1

Nro. da Conta: 54321
Saldo da Conta: -100.00
Deseja entra mais dados (0:Não, 1:Sim) ? 1

Nro. da Conta: 11111
Saldo da Conta: 150.00
Deseja entra mais dados (0:Não, 1:Sim) ? 1

Nro. da Conta: 22222
Saldo da Conta: -150.00
Deseja entra mais dados (0:Não, 1:Sim) ? 2
Opção inválida. Deseja entra mais dados (0:Não, 1:Sim) ? 0

>> Fim da Entrada de Dados <<
```

**Etapa 2. Atendimento de clientes: Saques/Pagamentos (saída) e Depósitos (entrada):** Uma vez que os dados da situação inicial das contas foram lidos, o programa irá entrar no modo de atendimento de clientes (guichê de banco). Neste modo, o programa pede o código da conta do usuário que está sendo atendido, localiza o cadastro desta conta, e solicita para o cliente indicar se vai realizar um crédito (depósito) ou um débito (saque), indicando o valor correspondente da operação. O programa deve atualizar o saldo atual da conta do cliente. Se a conta não for encontrada, o usuário é avisado e a operação não é completada. Esta etapa do programa de atendimento a clientes fica neste laço de saques e depósitos até que o usuário indique um código especial de conta (**código -1**) que serve para indicar quando o programa deve passar para a etapa seguinte (Etapa 3). **ATENÇÃO LEIA A ETAPA 3 ANTES DE IMPLEMENTAR A ETAPA 2 [3.0 pontos]**

Exemplo da tela atendimento de clientes (saques/depósitos):

```
>> Atendimento de Clientes (Digite -1 para encerrar) <<
```

```
Informe o Código da Conta: 12345
```

```
Tipo da Operação (S=Saque, D=Depósito): S
```

```
Valor da operação: 30.00
```

```
Saldo Atual: 70.00
```

```
Informe o Código da Conta: 12345
```

```
Tipo da Operação (S=Saque, D=Depósito): S
```

```
Valor da operação: 45.00
```

```
Saldo Atual: 25.00
```

```
Informe o Código da Conta: 11111
```

```
Tipo da Operação (S=Saque, D=Depósito): S
```

```
Valor da operação: 200.00
```

```
Saldo Atual: -50.00
```

```
Informe o Código da Conta: 22222
```

```
Tipo da Operação (S=Saque, D=Depósito): D
```

```
Valor da operação: 200.00
```

```
Saldo Atual: 50.00
```

```
Informe o Código da Conta: 33333
```

```
Erro: Código de Conta Inexistente!
```

```
Informe o Código do Conta: -1
```

```
>> Fim do Atendimento de Clientes <<
```

**Etapa 3. Consolidação dos dados ao final do dia, listando as contas e o saldo atual (balancete):** Uma vez que as atividades do dia foram encerradas, o programa irá fechar o balanço final das contas e da movimentação do dia. Exibir na tela:

- 1) Para cada conta: exibir o número da conta e o seu respectivo saldo, destacando se for negativo;
- 2) O valor total de depósitos realizados no dia;
- 3) O valor total de saques realizados no dia;
- 4) O balanço (fluxo de caixa total) totalizando depósitos e saques.

[2.0 pontos]

Exemplo da tela com os totais:

>> Balanço do Movimento Bancário <<

Código da Conta: 12345 - Saldo Atual: R\$ 25.00  
Código da Conta: 54321 - Saldo Atual: R\$ -100.00 (CONTA NEGATIVA)  
Código da Conta: 11111 - Saldo Atual: R\$ -50.00 (CONTA NEGATIVA)  
Código da Conta: 22222 - Saldo Atual: R\$ 50.00

Total de Depósitos: R\$ 200.00  
Total de Saques: R\$ 275.00  
Fluxo de Caixa: R\$ -75.00

>> FIM <<

⇒ A prova possui 3 etapas mas se constitui de um **único programa COMPLETO** que deve incluir estas três etapas em seqüência.

---

---

### **REGRAS EM RELAÇÃO REALIZAÇÃO DESTA PROVA**

1. A PROVA É **INDIVIDUAL**.
2. A PROVA É **COM CONSULTA AO MATERIAL INDIVIDUAL**.  
(Pode consultar: cadernos, anotações, livros – qualquer tipo de material escrito ou impresso)
3. **NÃO É PERMITIDO O EMPRÉSTIMO DE MATERIAL** (Cadernos, Anotações, Livros, etc).
4. **NÃO É PERMITIDO O USO DE DISPOSITIVOS ELETRÔNICOS** durante a prova.  
(Não pode usar: notebook, computador, palmtops/pdas, celular, etc.)
5. RESPONDER A PROVA NAS FOLHAS FORNECIDAS: A CANETA OU A LÁPIS.  
**SE FOR RESPONDIDA A LÁPIS E TIVER QUALQUER INDÍCIO DE ALTERAÇÃO OU RASURA, NÃO SERÃO ACEITOS PEDIDOS DE REVISÃO DE PROVA.**
6. LEMBRE-SE DE **IDENTIFICAR A PROVA** COM O SEU NOME E NÚMERO USP.  
DEVOLVER A FOLHA DE RESPOSTAS JUNTAMENTE COM A PROVA (Questões).
7. DURAÇÃO: 3 Horas

>> *Não será tolerado qualquer tipo de troca de informações entre alunos durante a prova!*

>> *Responda a prova de forma LEGÍVEL. Se não for possível entender o texto, é questão errada!*

>> *Qualquer dúvida, chame o professor e aguarde SENTADO em sua cadeira para ser atendido.*

>> *Somente se dirija ao professor quando for entregar a prova.*

---

---

FIM

---

---