

Nome: _____ 2º semestre 2014

Instruções para a prova

- A prova é sem consulta;
- A prova dura 1 hora e 40 minutos;
- Esta folha de enunciados deverá ser entregue ao professor junto com a folha de respostas;
- Onde for adequado, use a função `float pow(float x, float y)` para calcular x^y , a função `float sqrt(float x)` para calcular \sqrt{x} , a função `float cbrt(float x)` para calcular $\sqrt[3]{x}$, a função `int abs(int x)` para calcular o valor absoluto (módulo) de um número inteiro x , e a função `float fabsf(float y)` para calcular o valor absoluto (módulo) de um número real y .
- Nos exemplos de execução de programas, a saída para a tela emitida pelo programa está em *itálico* e a entrada do usuário está representada em **negrito**.

Questão 1 (50 pontos)

Faça um programa em C++ que permita consultar a classificação de equipes no mundial de construtores de automobilismo. Cada equipe possui dois pilotos: um piloto principal e um piloto secundário. Considere dois vetores para representar esses pilotos: (i) vetor com a pontuação dos pilotos principais de cada equipe e (ii) vetor com a pontuação dos pilotos secundários de cada equipe. Considere também que pilotos da mesma equipe estão em posições opostas em cada vetor. Por exemplo: o primeiro piloto principal e o último piloto secundário são da mesma equipe, o segundo piloto principal e o penúltimo piloto secundário são de outra equipe, e assim por diante. A pontuação dos pilotos não está necessariamente ordenada.

O número N de pares de pilotos que disputam o campeonato é fixo e deve ser informado via `#define`. Seu programa deve obter a pontuação de cada piloto principal, de cada piloto secundário e calcular a pontuação de cada equipe. Guarde a pontuação de sua respectiva equipe no lugar da pontuação original de cada piloto. Mostre como está a classificação das equipes, imprimindo as posições dos dois vetores. Mostre também quem é a equipe que lidera o campeonato.

OBS.: Para a exibição do conteúdo do vetor na tela, considere que já existe a função `void imprime_vetor (int vet[])`. Sua solução deve apenas chamar esta função.

Exemplo de execução (para N=3):

```
Pontos p/ piloto principal: 5 2 4
Pontos p/ piloto secundário: 3 6 9
Pontuacao por equipes:
14 8 7
7 8 14
Equipe 1 na lideranca
```

Outro exemplo de execução (para N=5):

```
Pontos p/ piloto principal: 5 2 4 6 3
Pontos p/ piloto secundario: 1 9 0 10 8
Pontuacao por equipes:
13 12 4 15 4
4 15 4 12 13
Equipe 4 na lideranca
```

Questão 2 (50 pontos)

Um comerciante deseja catalogar os M produtos que comercializa em uma matriz de M linhas. Para cada produto, ele deseja registrar como colunas (N) da matriz: o código, a quantidade em estoque, o custo unitário de aquisição e o preço unitário de venda praticado. Escrever um programa em C++ que permita ao comerciante desenvolver o cadastro desejado e, a partir deste, determinar o custo total de seu estoque, o item com maior rentabilidade absoluta (obtida pela diferença entre preço e custo unitários) e sua perspectiva de ganho se vender todo o estoque deste item. Os valores de M e N deverão ser estabelecidos via diretiva `#define`.

OBS.: Para a leitura de uma matriz, considere que já existe a função `void ler_matriz (float matriz[][N])`. Sua solução deve apenas CHAMAR esta função.

Exemplo de execução (com M=8, N=4):

```
CATÁLOGO DE PRODUTOS
Cód.      Estoque  Custo  Preço
.00010    35000    .025   .05
.01023     135     7.134   9.5
.23189    3345     2.74    5
.00235     15     115.285 209
.12131   52356     .3598   .85
.00123     854     12.557 13.3
.00020    2600     .472    1
.00025    1000     .125    .50
Custo total do estoque: 43646.2318
Item com maior rentabilidade: .00235
Lucro bruto previsto: 1405.725
```

Nome: _____ 2º semestre 2014

Instruções para a prova

- A prova é sem consulta;
- A prova dura 1 hora e 40 minutos;
- Esta folha de enunciados deverá ser entregue ao professor junto com a folha de respostas;
- Onde for adequado, use a função `float pow(float x, float y)` para calcular x^y , a função `float sqrt(float x)` para calcular \sqrt{x} , a função `float cbrt(float x)` para calcular $\sqrt[3]{x}$, a função `int abs(int x)` para calcular o valor absoluto (módulo) de um número inteiro x , e a função `float fabsf(float y)` para calcular o valor absoluto (módulo) de um número real y .
- Nos exemplos de execução de programas, a saída para a tela emitida pelo programa está em *itálico* e a entrada do usuário está representada em **negrito**.

apenas a primeira que satisfaz a condição.

OBS.: Para a leitura de uma matriz, considere que já existe a função `void ler_matriz (int matriz[][N])`. Sua solução deve apenas CHAMAR esta função.

Exemplo de execução (com M=4 e N=3):

Informe a matriz:
8 2 7
4 5 5
3 10 2
5 10 1
linha 0, soma 17

Questão 1 (50 pontos)

Faça um programa em C++ que leia (do teclado) um vetor de tamanho N (N estabelecido via diretiva `#define`) de números reais e crie um novo vetor, de mesmo tamanho e tipo do original, com as seguintes características: **(a)** valores dos elementos das extremidades do novo vetor (primeiro e último) idênticos aos do original; **(b)** valores dos elementos do “miolo” do vetor (segundo ao penúltimo) obtidos a partir da média dos valores de seu antecessor e sucessor posicionais no vetor original, desde que seu valor esteja situado acima do valor de seu antecessor e abaixo do valor de seu sucessor no vetor original; não verificada esta característica, gerar o elemento do vetor de saída com o mesmo valor do original.

Ao final do processamento, o programa deverá mostrar o vetor gerado no monitor de vídeo e informar quantas substituições de valores foram feitas.

OBS.: Considere que a função `void imprime_vetor(float vet[])` para mostrar o vetor criado JÁ EXISTE. Sua solução deverá apenas CHAMAR esta função onde necessário.

Exemplo de execução (com N=10):

Informe vetor:
3 4.5 2 5 6 7.3 7.5 9 12 15
Vetor gerado:
3 4.5 2 4 6.15 6.75 8.15 9.75 12 15
Alterações: 6

Questão 2 (50 pontos)

Faça um programa em C++ que leia uma matriz $M \times N$ de números inteiros (M e N definidos via diretiva `#define`) e calcula qual é a linha cuja soma é maior que a soma das outras linhas, informando este valor ao usuário. Caso haja mais que uma linha com soma maior que as outras, mostrar

Nome: _____ 2º semestre 2014

Instruções para a prova

- A prova é sem consulta;
- A prova dura 1 hora e 40 minutos;
- Esta folha de enunciados deverá ser entregue ao professor junto com a folha de respostas;
- Onde for adequado, use a função `float pow(float x, float y)` para calcular x^y , a função `float sqrt(float x)` para calcular \sqrt{x} , a função `float cbrt(float x)` para calcular $\sqrt[3]{x}$, a função `int abs(int x)` para calcular o valor absoluto (módulo) de um número inteiro x , e a função `float fabsf(float y)` para calcular o valor absoluto (módulo) de um número real y .
- Nos exemplos de execução de programas, a saída para a tela emitida pelo programa está em *itálico* e a entrada do usuário está representada em **negrito**.

Questão 1 (50 pontos)

Faça um programa em C++ que leia um vetor A de 6 elementos contendo o gabarito da Mega Sena. A seguir, ler um vetor B de 10 elementos contendo uma aposta. Escrever quantos pontos fez o apostador, e se ele fez a sena (6 acertos), a quina (5 acertos) ou a quadra (4 acertos). E no vetor B , mudar o valor do elemento que teve acerto para 99.

OBS.: Considere que a função `void imprime_vetor(int vet[])` para mostrar o vetor criado JÁ EXISTE. Sua solução deverá apenas CHAMAR esta função onde necessário.

Exemplo de execução:

```
Vetor A:  7 43 51 47 6 3
Vetor B:  7 43 9 1 5 8 22 47 6 34
Fez a quadra
99 99 9 1 5 8 22 99 99 34
```

Questão 2 (50 pontos)

O gabarito de cada participante de uma prova de concurso foi digitalizado e uma matriz de números inteiros correspondente foi gerada. Essa matriz tem uma linha para cada questão e uma coluna para cada alternativa. Assim, cada posição contém o valor 0 se a alternativa não foi marcada pelo participante e 1 caso contrário. Faça um programa em C++ que leia um vetor de tamanho N , onde cada posição i contém um `char` representando a alternativa correta da i -ésima questão ('A' se a primeira alternativa é a correta, 'B' se a segunda, e assim por diante). Em seguida, seu programa deve ler uma matriz de dimensões $N \times M$ representando as repostas de um participante e, por fim, deve imprimir a quantidade de questões que o participante acertou. Considere que questões com 2 ou mais alternativas marcadas estão **incorretas**. Os valores N e M devem ser estabelecidos via `#define`.

Dica: O código ASCII do caractere 'A' é 65, do 'B' é 66, e assim por diante.

OBS.: Para a leitura da matriz e do vetor, considere que já existem as funções `void le_matriz(int mat[][M])` e `void le_vetor(int vet[])`. Seu programa deve apenas CHAMAR estas funções.

Exemplo de execução (com $N=5$, $M=4$):

Entre com o vetor:

A A D B C

Entre com o gabarito:

0 0 1 0

1 0 0 0

1 0 0 1

0 1 0 0

0 0 0 1

O participante acertou 2 questões.

p3_2014s2_sols

```
//----- 1-----
#include<iostream>
#include<iomanip>
#define N 3
using namespace std;
int lev(int xp[N], int xs[N]){
    int i;
    for (i=0;i<N;i++){
        cout<<"Informe pontuacao piloto principal - ";
        cin>>xp[i];
    }
    for (i=0;i<N;i++){
        cout<<"Informe pontuacao piloto reserva - ";
        cin>>xs[i];
    }
}
int main() {
    int vp[N];
    int vs[N];
    int i,aux,maximo,imaximo;
    lev(vp,vs);
    for (i=0;i<N;i++){
        aux=vp[i]+vs[(N-i)-1];
        vp[i]=aux;
        vs[(N-i)-1]=aux;
    }
    cout<<"Pontuacao por equipes "<<endl;
    for (i=0;i<N;i++){
        cout<<setw(4)<<vp[i]<<' ';
    }
    cout<<endl;
    for (i=0;i<N;i++){
        cout<<setw(4)<<vs[i]<<' ';
    }
    cout<<endl;
    maximo=-99999;
    for (i=0;i<N;i++){
        if (vp[i]>maximo){
            maximo=vp[i];
            imaximo=i+1;
        }
    }
    cout<<"Equipe "<<imaximo<<" na lideraca"<<endl;
}
//-----
#include<iostream>
#include<iomanip>
#define M 8
#define N 4
using namespace std;

int main() {
    float es[M][N]={{0.00010,35000,0.025,0.05},
                    {0.01023,135,7.134,9.5},
                    {0.23189,3345,2.74,5},
                    {0.00235,15,115.285,209},
                    {0.12131,52356,0.3598,0.85},
                    {0.00123,854,12.557,13.3},
                    {0.00020,2600,0.472,1},
                    {0.00025,1000,0.125,0.50}};
    float cte,mra,pg,mai;
```

```

int i,imai;
cte=0;
for (i=0;i<M;i++){
    cte=cte+es[i][1]*es[i][2];
}
mai=-99999;
for (i=0;i<M;i++){
    if((es[i][3]-es[i][2])>mai){
        mai=es[i][3]-es[i][2];
        imai=i;
    }
}
pg=es[imai][1]*(es[imai][3]-es[imai][2]);
cout<<"Custo total do estoque "<<cte<<endl;
cout<<"item com maior lucratividade "<<es[imai][0]<<endl;
cout<<"Lucro bruto previsto "<<pg<<endl;
}
//-----
#include<iostream>
#include<iomanip>
#define N 10
using namespace std;
int main() {
//    float v[N]={3,4.5,2,5,6,7.3,7.5,9,12,15};
    float v[N];
    float vs[N];
    int i,ct;
    ct=0;
    cout<<"Informe vetor "<<endl;
    for (i=0;i<N;i++){
        cin>>v[i];
    }
    vs[0]=v[0];
    vs[N-1]=v[N-1];
    for (i=1;i<N-1;i++){
        if ((v[i]>v[i-1])&&(v[i]<v[i+1])){
            vs[i]=(v[i-1]+v[i+1])/2;
            ct++;
        }
        else{
            vs[i]=v[i];
        }
    }
    for (i=0;i<N;i++){
        cout<<setw(4)<<vs[i]<<' ';
    }
    cout<<endl<<"Alteracoes "<<ct;
}
//-----
#include<iostream>
#include<iomanip>
#define M 4
#define N 3
using namespace std;
int main() {
    int m[M][N]={{8,2,7},{4,5,5},{3,10,2},{5,10,1}};
    int som,msom,imsom;
    msom=-99999;
    int i,j;
    for (i=0;i<M;i++){
        som=0;
        for(j=0;j<N;j++){
            som=som+m[i][j];

```

```

    }
    if (som>msom){
        msom=som;
        imsom=i;
    }
}
cout<<"linha "<<imsom<<" ,   soma"<<msom<<endl;
}
//-----
#include<iostream>
#include<iomanip>
#define M 4
#define N 3
using namespace std;
int main(){
    int a[6];
    int b[10];
    int i,j,ac;
    cout<<"Vetor A: ";
    for(i=0;i<6;i++){
        cin>>a[i];
    }
    cout<<"Vetor B: ";
    for(i=0;i<10;i++){
        cin>>b[i];
    }
    ac=0;
    for(i=0;i<6;i++){
        for(j=0;j<10;j++){
            if (a[i]==b[j]){
                ac++;
                b[j]=99;
            }
        }
    }
    if (ac==6){
        cout<<"Fez a sena"<<endl;
    }
    if (ac==5){
        cout<<"Fez a quina"<<endl;
    }
    if (ac==4){
        cout<<"Fez a quadra"<<endl;
    }
    for(i=0;i<10;i++){
        cout<<b[i]<<' ';
    }
}
//-----

#include<iostream>
#define N 5 //numero de linhas
#define M 4 // quantidade de alternativas
using namespace std;
int main(){
    int mat[N][M]={{0,0,1,0},{1,0,0,0},{1,0,0,1},{0,1,0,0},{0,0,0,1}};
    char gab[]="AADBC";
    int i,j,som,qual,nota=0;
    for (i=0;i<N;i++){
        som=0;
        for (j=0;j<M;j++){
            som=som+mat[i][j];
        }
    }
}

```

```
qual=gab[i]-65;
if ((som==1)&&(mat[i][qual]==1)){
    nota=nota+1;
}
}
cout<<"O participante acertou "<<nota<<" questoes";
}
```