**TAREFA DE FIXAÇÃO / AVALIAÇÃO**

a) Faça o algoritmo para calcular o valor de uma prestação em atraso com base na multa de 2% e também do juros mensal de 1% sobre o valor principal. Declare vetores explícitos para entrada do valor da prestação, valorprincipal [ ] e da quantidade de dias em atraso: dias [ ]. Armazene os resultados no vetor ValorPagar [ 5 ]. Ao final imprima a tabela com os vetores valorPrincipal [], dias [] e valorpagar[]. Faça um menu de controle.

**Código Fonte:**

#include "iostream"

#include "iomanip"

#include "cstdlib"

using namespace std;

double valorprincipal[] = {1200, 2530, 50000, 2100, 3250};

int dias[] = {10, 30, 16, 62, 5};

double ValorPagar[5];

int tamanhoValor = sizeof(valorprincipal) / sizeof(double);

void calcularValor(){

int mes = 0;

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

{

mes = dias[i] / 30;

ValorPagar[i] = valorprincipal[i] \* 1.02;

if (mes >= 1){

ValorPagar[i] += valorprincipal[i] \* (0.01 \* mes);

}

}

}

void mostrar(){

const int tValor = 14;

const int tDias = 6;

const char espaco = ' ';

calcularValor();

cout << "\n|" << left << setw(tValor) << setfill(espaco) << " V. PRINCIPAL" << "|";

cout << left << setw(tDias) << setfill(espaco) << " DIAS" << "|";

cout << left << setw(tValor) << setfill(espaco) << " V. PAGAR" << "|" << endl;

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

{

cout << "|";

cout << left << setw(tValor) << setfill(espaco) << valorprincipal[i] << "|";

cout << left << setw(tDias) << setfill(espaco) << dias[i] << "|";

cout << left << setw(tValor) << setfill(espaco) << ValorPagar[i] << "|" << endl;

}

}

int main(){

int opcao = 0;

system("CLS");

MENU:

cout << "\nEscolha uma opcao\n1. Mostrar Valores\n0. Fechar Programa\n> ";

cin >> opcao;

switch (opcao)

{

case 1:

mostrar();

break;

case 0:

system("CLS");

cout << "O programa foi encerrado!" << endl;

exit(0);

break;

default:

cout << "Opcao nao reconhecida" << endl;

break;

}

goto MENU;

return 0;

}

**Tela de Execução:**

Texto

Descrição gerada automaticamente



b) Crie um algoritmo que tenha um vetor explícito para armazenar 5 idades e um vetor não explícito para armazenar 5 nomes. Ao final Exiba o nome da pessoa de maior idade, menor idade, a média das idades e o saldo das idades.

**Código Fonte:**

#include "iostream"

using namespace std;

int idades[] = {18, 17, 19, 20, 25};

string nomes[5] = {};

int maior = 0, menor = 0, posMaior = 0, posMenor = 0;

int tamanhoNome = sizeof(nomes)/sizeof(string);

int tamanhoIdade = sizeof(idades)/sizeof(int);

string lerNome(){

string nome;

cout << "\nInsira um nome: " << endl;

getline(cin, nome);

return nome;

}

void acharMaiorMenor(){

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

{

if (idades[i] > maior || i == 0) {

maior = idades[i];

posMaior = i;

}

if (idades[i] < menor || i == 0) {

menor = idades[i];

posMenor = i;

}

}

}

int acharSaldo(){

int saldo = 0;

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

{

saldo += idades[i];

}

return saldo;

}

float acharMedia(){

float media = 0;

media = acharSaldo();

media /= tamanhoIdade;

return media;

}

void mostrar(){

float media = 0;

int saldo = 0;

acharMaiorMenor();

media = acharMedia();

saldo = acharSaldo();

cout << "\nPessoa com a maior idade: " << nomes[posMaior] << ", " << maior << endl;

cout << "Pessoa com a menor idade: " << nomes[posMenor] << ", " << menor << endl;

cout << "Media das idades: " << media << endl;

cout << "Saldo das Idades: " << saldo << endl;

}

int main(int argc, char const \*argv[])

{

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

{

nomes[i] = lerNome();

}

mostrar();

return 0;

}

**Tela de Execução:**

Tela de computador com texto preto sobre fundo branco

Descrição gerada automaticamente