**Nome: Bruno Deluca Satil Cassiano**

Exercício 1 (bem fácil só para começar)

Faça um programa onde o usuário insira um número positivo então esse programa deve imprimir todos os números entre o número inserido e 0.

#include <iostream>

using namespace std;

void contarNumerosDoIntervalo(int numeroInicial);

int num;

int main()

{

cout<<"Insira o número inicial: ";

cin>>num;

contarNumerosDoIntervalo(num-1);

return 0;

}

void contarNumerosDoIntervalo(int numeroInicial){

while (numeroInicial > 0){

cout<<numeroInicial <<endl;

numeroInicial--;

}

}

Exercício 2

* Utilizando funções, faça um programa de uma calculadora que faça as 4 operações fundamentais da matemática.

#include <iostream>

#include <iomanip>

using namespace std;

void calculadora(double num1, double num2);

double num1;

double num2;

int main()

{

cout<<"Insira o primeiro número: ";

cin>>num1;

cout<<endl;

cout<<"Insira o segundo número: ";

cin>>num2;

cout<<endl;

calculadora(num1, num2);

return 0;

}

void calculadora(double num1, double num2) {

cout<< fixed <<setprecision(2) <<"Soma: " <<num1 + num2 <<endl;

cout<< fixed <<setprecision(2) <<"Subtração: " <<num1 - num2 <<endl;

cout<< fixed <<setprecision(2) <<"Multiplicação: " <<num1 \* num2 <<endl;

cout<< fixed <<setprecision(2) <<"Divisão: " <<num1 / num2 <<endl;

}

Exercício 3

* Faça um programa onde o usuário insira 3 números, e através de uma função o programa deve dizer qual é o maior número e qual é o menor número.

#include <iostream>

using namespace std;

int menor(int vetor[3]);

int maior(int vetor[3]);

int vetor[3];

int count = 0;

int main()

{

for (int count = 0; count < 3; count++) {

cout<<"Insira o "<<count + 1 <<"º número: ";

cin>>vetor[count];

cout<<endl <<endl;

}

cout<<"O maior é o número " <<maior(vetor) <<endl;

cout<<"O menor é o número " <<menor(vetor) <<endl;

return 0;

}

int menor(int vetor[3]) {

int menor = vetor[0];

for (count = 0; count < 3; count++) {

if (vetor[count] < menor) {

menor = vetor[count];

}

}

return menor;

}

int maior(int vetor[3]) {

int maior = vetor[0];

for (count = 0; count < 3; count++) {

if (vetor[count] > maior) {

maior = vetor[count];

}

}

return maior;

}

Exercício 4

* Faça um programa onde o usuário insira 10 números que serão armazenados em um vetor, logo após, crie 3 funções, uma função que ordene os 10 números em ordem crescente, uma função que ordene os 10 números em ordem decrescente e uma função que imprima o vetor. Faça esse programa rodar quantas vezes o usuário quiser (looping infinito) com opção do usuário de sair quando desejar.

#include <iostream>

using namespace std;

int vetor[10];

int count = 0;

void ordemDecrescente(int vetor[10]);

int vetorDecrescente[10];

void ordemCrescente(int vetor[10]);

int vetorCrescente[10];

void ordemNormal(int vetor[10]);

int countExterno = 0;

int countInterno = 0;

int holder = 0;

int loop = 0;

int main()

{

while (loop == 0) {

for(count = 0; count < 10; count++) {

cout<<"Insira o " <<count + 1 << "º número do vetor: ";

cin>>vetor[count];

cout<<endl <<endl;

}

ordemNormal(vetor);

ordemDecrescente(vetor);

ordemCrescente(vetor);

cout<<"Insira 0 para o programa continuar e qualquer outro número para o programa encerrar: ";

cin>>loop;

cout<<endl <<endl;

}

return 0;

}

void ordemCrescente(int vetor[10]) {

for (countExterno = 0; countExterno < 10;) {

for(countInterno = 0; countInterno < 10; countInterno++) {

if (vetor[countExterno] < vetor[countInterno]) {

holder = vetor[countExterno];

vetor[countExterno] = vetor[countInterno];

vetor[countInterno] = holder;

}

}

countExterno++;

}

cout<<"Vetor em ordem crescente: ";

for(count = 0; count < 10; count++) {

cout<<"[" <<vetor[count] <<"] ";

};

cout<<endl <<endl;

}

void ordemDecrescente(int vetor[10]) {

for (countExterno = 0; countExterno < 10;) {

for(countInterno = 0; countInterno < 10; countInterno++) {

if (vetor[countExterno] > vetor[countInterno]) {

holder = vetor[countExterno];

vetor[countExterno] = vetor[countInterno];

vetor[countInterno] = holder;

}

}

countExterno++;

}

cout<<"Vetor em ordem decrescente: ";

for(count = 0; count < 10; count++) {

cout<<"[" <<vetor[count] <<"] ";

};

cout<<endl <<endl;

}

void ordemNormal(int vetor[10]) {

cout<<"Vetor em ordem normal: ";

for(count = 0; count < 10; count++) {

cout<<"[" <<vetor[count] <<"] ";

};

cout<<endl <<endl;

}

Exercício 5

* Faça um programa onde leia uma nota de 0 a 10, utilize uma função verificação para ter certeza que o usuário inseriu uma nota dentro desses valores (utilize uma estrutura de repetição para só sair do looping quando o usuário inserir uma nota válida).
* Através de uma segunda função, faça com que o programa entenda a nota que foi inserida e a insira em um dos 3 cenários (entre 10 a 7 nota A, entre 6,9 a 4 nota B, menor que 4, nota C).
* Por último, através de uma terceira função, faça com que o programa pegue a letra que corresponde a nota como argumento e imprima uma mensagem na tela sendo A=aprovado, B= recuperação, C= reprovado.

#include <iostream>

#include <string>

using namespace std;

int nota = -1;

bool verificacao(double nota);

bool verificador = false;

char notaValorParaLetra(double nota);

char notaLetra = ' ';

string mensagemDoResultado(char notaLetra);

string mensagem = " ";

int main()

{

while (verificador == false) {

cout<<"Insira uma nota no intervalo de 0 a 10: ";

cin>>nota;

cout<<endl;

verificador = verificacao(nota);

}

notaLetra = notaValorParaLetra(nota);

mensagem = mensagemDoResultado(notaLetra);

cout<<mensagem;

return 0;

}

bool verificacao(double nota) {

if (!(nota >= 0 && nota <= 10)) {

cout<<"Número inválido, tente novamente." <<endl <<endl;

return false;

} else {

return true;

}

}

char notaValorParaLetra(double nota) {

if (nota < 4) {

return 'C';

} else if (nota >= 4 && nota <= 6.9) {

return 'B';

} else {

return 'A';

}

}

string mensagemDoResultado(char notaLetra) {

if (notaLetra == 'A') {

return "Aprovado";

} else if (notaLetra == 'B') {

return "Recuperação";

} else if (notaLetra == 'C') {

return "Reprovado";

} else {

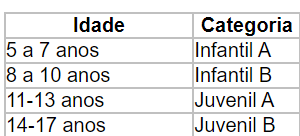
return "Erro Identificado!";

}

}

Exercício 6

* Faça um programa para um campeonato de natação. Onde o usuário deva inserir o sexo do competidor e a idade. Além de uma função para verificar se a idade está entre 5 a 17 anos, é preciso utilizar mais 1 função para cada uma das validações (idade e sexo):  
  Sexo se for homem ou mulher e idade seguindo a tabela:



Com os 2 retornos da função, faça com que o programa entenda esses retornos e fale em qual campeonato essa pessoa se encaixe a impressão deve ser algo similar a ex: Feminino Infantil B

#include <iostream>

#include <string>

using namespace std;

string sexo;

bool verificarSexo(string sexo);

string categoriaGenero;

int idade;

bool verificarIdade(int idade);

bool verificador = false;

void categoriaMensagemFinal(string categoriaGenero, int idade);

int main()

{

while (verificador == false) {

cout<<"Você é homem ou mulher? ";

cin>>sexo;

cout<<endl;

verificador = verificarSexo(sexo);

}

verificador = false;

while (verificador == false) {

cout<<"Qual sua idade? ";

cin>>idade;

cout<<endl;

verificador = verificarIdade(idade);

}

categoriaMensagemFinal(categoriaGenero, idade);

return 0;

}

bool verificarSexo(string sexo){

if (sexo == "homem") {

categoriaGenero = "Masculino";

return true;

} else if (sexo == "mulher") {

categoriaGenero = "Feminino";

return true;

} else {

cout<<"Sexo inválido. Neste exercício estamos considerando apenas homem ou mulher." <<endl <<endl;

return false;

}

}

bool verificarIdade(int idade){

if (idade >= 5 && idade <= 17) {

return true;

} else {

cout<<"Os alunos devem ter entre 5 a 17 anos." <<endl <<endl;

return false;

}

}

void categoriaMensagemFinal(string categoriaGenero, int idade) {

if (idade >= 5 && idade <= 7) {

cout<<categoriaGenero <<" Infantil A";

} else if (idade >= 8 && idade <= 10) {

cout<<categoriaGenero <<" Infantil B";

} else if (idade >= 11 && idade <= 13) {

cout<<categoriaGenero <<" Juvenil A";

} else if (idade >= 14 && idade <= 17) {

cout<<categoriaGenero <<" Juvenil B";

}

}

Exercício 7 (longo e difícil RMK)

* Faça um programa para um joguinho de tiro. Nesse jogo, o jogador deve escolher uma entre 3 armas e elas tem uma capacidade fixa de munição:

Revolver = 6 balas  
Escopeta = 8 balas  
Rifle = 10 balas

Primeiramente o jogador deve escolher uma arma, escolher atirar, trocar de arma ou recarregar. O jogador não poderá atirar se a munição da arma selecionada estiver acabada. Utilize uma função para atirar, uma função para recarregar e você também (pode utilizar se achar melhor) uma função para trocar de arma.

#include <iostream>

using namespace std;

enum {

REVOLVER = 6,

ESCOPETA = 8,

RIFLE = 10

};

int balas[3];

int pente = 0;

int opcao = 0;

int arma = 0;

int armaEmUso = 0;

string armaEquipada[3];

bool prontoProCombate = true;

void selecionarArma();

void trocarArma();

void recarregarArma();

void iniciandoJogo();

int count = 0;

int main()

{

iniciandoJogo();

selecionarArma();

while (prontoProCombate == true) {

cout<<"Escolha a opção correspondente a ação que você quer tomar a seguir:"<<endl;

cout<<"1 - Trocar de arma" <<endl;

cout<<"2 - Atirar" <<endl;

cout<<"3 - Recarregar" <<endl;

cin>>opcao;

cout<<endl <<endl;

switch (opcao) {

case 1:

selecionarArma();

break;

case 2:

if (balas[arma-1] == 0) {

cout<<"Você estava sem balas, então quando tentou atirar, apenas revelou sua posição e foi morto!" <<endl <<endl;

prontoProCombate = false;

} else {

balas[arma-1]--;

cout<<"BANG! Headshot! Agora "<<armaEquipada[arma-1] <<" está com " <<balas[arma-1] <<" balas." <<endl <<endl;

}

break;

case 3:

recarregarArma();

break;

}

}

return 0;

}

/\*

------------------------------------------------------------------------------------------------------

------------------------------------------------------------------------------------------------------

------------------------------------------------------------------------------------------------------

\*/

void iniciandoJogo(){

balas[0] = REVOLVER;

balas[1] = ESCOPETA;

balas[2] = RIFLE;

armaEquipada[0] = "o revolver";

armaEquipada[1] = "a escopeta";

armaEquipada[2] = "o rifle";

}

void selecionarArma() {

armaEmUso = arma;

arma = 0;

while (arma != 1 && arma != 2 && arma != 3) {

trocarArma();

if (arma != armaEmUso && (arma == 1 || arma == 2 || arma == 3)) {

pente = balas[arma-1];

cout<<"Você equipou " <<armaEquipada[arma-1] <<". O pente está carregado com " <<pente <<" balas." <<endl <<endl;

} else if (arma == 4 || arma == armaEmUso) {

cout<<"Arma mantida!" <<endl <<endl;

arma = armaEmUso;

} else {

cout<<"Só há 3 opções de arma e a opção de não trocar. Escolha novamente." <<endl <<endl;

}

}

}

void trocarArma() {

if (armaEmUso != 0) {

cout<<"Escolha uma das outras armas ou mantenha a atual: " <<endl;

} else {

cout<<"Você tem 3 opções de arma, escolha a que preferir:" <<endl;

}

if (armaEmUso != 1) {

cout<<"1 - Revolver" <<endl;

}

if (armaEmUso != 2) {

cout<<"2 - Escopeta" <<endl;

}

if (armaEmUso != 3) {

cout<<"3 - Rifle" <<endl;

}

if (armaEmUso != 0) {

cout<<"4 - Não quero mais trocar de arma" <<endl;

}

cin>>arma;

cout<<endl <<endl;

}

void recarregarArma() {

switch (arma) {

case 1:

cout<<"Revolver recarregado e pronto pro Penta-Kill!" <<endl <<endl;

balas[0] = REVOLVER;

break;

case 2:

cout<<"Escopeta recarregada e pronta pro Penta-Kill!" <<endl <<endl;

balas[1] = ESCOPETA;

break;

case 3:

cout<<"Rifle recarregado e pronto pro Penta-Kill!" <<endl <<endl;

balas[2] = RIFLE;

break;

}

}