Задача. Дано масив цін товарів. Користувач вводить кількість наявних у нього грошей. Вивести на екран ціни тих товарів, які він може собі купити.

// ConsoleApplication19.cpp : Defines the entry point for the console application.

//

#include "stdafx.h"

#include <iostream>

using namespace std;

int main()

{

//Задача.Дано масив цін товарів.Користувач вводить кількість наявних у нього грошей.

//Вивести на екран ціни тих товарів, які він може собі купити.

//1.Вводимо кількість цін (кількість товарів)

int productsCount;

printf("Products count: ");

//2.Виділяємо пам"ять для масиву

double \* prices = new double[productsCount];

//3.Вводимо масив цін

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

{

printf("Proce of product %d:",i+1);

cin >> prices[i];

}

//4.Вводимо кількість грошей користувача

double money;

printf("Money:");

cin >> money;

//5.Перебираємо масив цін і виводимо ті, які менші за кількість грошей користувача

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

{

if (prices[i]<=money)

{

printf("Product - %d , price - %.2f",i+1,prices[i]);

}

}

system("pause");

return 0;

}

Задача. Дано масив з балами учнів за тест. Вивести на екран номери тих, які мають двійки (менше 60 балів).

Задача. Дано масив прибутків компанії кожного місяця за N років (згенерувати випадковим чином). Вивести на екран номери тих років, протягом яких був місяць з прибутком меншим за M.

Задача. Сформувати масив для гри «Сапер» (сформувати випадковим чином з 0 та 1). Користувач вводить номер рядка, номер стовпця і виграє, якщо там немає міни та програє в іншому випадку. Користувач робить не більше 5 спроб. Підрахувати кількість спроб до першого програшу.

// ConsoleApplication19.cpp : Defines the entry point for the console application.

//

#include "stdafx.h"

#include <iostream>

using namespace std;

int main()

{

//1.Вводимо кількість рядків і кількість стовпців мінного поля

int rowCount;

printf("Row count");

cin >> rowCount;

int columnCount;

printf("Column count:");

cin >> columnCount;

//2.Виділяємо пам"ять для двовимірного масиву

int\*\* mineField = new int\*[rowCount];

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

{

mineField[i] = new int[columnCount];

}

//3.Генеруємо масив випадковими числами (0 або 1)

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

{

for (int j = 0; j < columnCount; j++)

{

mineField[i][j] = rand() % 2;

}

}

//--------

printf("------------ MINE field -----------\n");

for (int j = 0; j < columnCount; j++)

{

printf("-----");

}

printf("\n");

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

{

for (int j = 0; j < columnCount; j++)

{

printf("| %d |", mineField[i][j]);

}

printf("\n");

for (int j = 0; j < columnCount; j++)

{

printf("-----");

}

printf("\n");

}

//4. 5 разів пробуємо кудисть ступати

int stepCountBeforeBoom = 0;

for (int i = 0; i < 5; i++)

{

//Вводимо координати клітинки

int row;

printf("row: ");

cin >> row;

int col;

printf("col: ");

cin >> col;

//Перевіряємо чи є там міна

if (mineField[row][col]==1)

{

break;

}

stepCountBeforeBoom++;

}

if (stepCountBeforeBoom == 5)

printf("You are lucky!!!\n");

else

printf("You are loooooooooser!\n");

system("pause");

return 0;

}