#include <iostream>

#include <cmath>

using namespace std;

class type1 {

public:

void Get\_answer() {

cout << "Ответом является любое значение Х" << endl;

}

void show() {

cout << "Уравнение 0 = 0" << endl;

}

};

class type2 {

private:

double A;

public:

type2(double a1) {

A = a1;

}

void Get\_answer() {

cout << "Ответом является Х = 0" << endl;

}

void show() {

cout << "Уравнение " << A << "\*x^2 = 0" << endl;

}

};

class type3 {

private:

double C;

public:

type3(double c1) {

C = c1;

}

void Get\_answer() {

cout << "Корней нет" << endl;

}

void show() {

cout << C << " = 0" << endl;

}

};

class type4 {

private:

double A, C;

public:

type4(double a1, double c1) {

A = a1;

C = c1;

}

void Get\_answer() {

double k = ((-1) \* C) / A;

if (k >= 0) {

double x1 = sqrt(k);

double x2 = sqrt(k) \* (-1);

cout << "Корни уравнения: " << endl << "X1 = " << x1 << endl << "X2 = " << x2 << endl;

}

else {

cout << "Корней нет" << endl;

}

}

void show() {

cout << "Уравнение " << A << "\*x^2 + " << C << " = 0" << endl;

}

void Get\_answer\_podbor() {

if ((((-1) \* C) / A) < 0) {

cout << "Корней методом подбора нет" << endl;

}

else {

double min = abs(0 - A \* (-20) \* (-20) - C);

double x = -20;

for (double i = -20; i < 20.01; i = i + 0.01) {

double temp = abs(0 - A \* (i) \* (i)-C);

if (temp < min) {

min = temp;

x = i;

}

}

cout << "Приближенно, один из корней уравнения: " << x << endl;

}

}

};

class type5 {

private:

double B, C;

public:

type5(double b1, double c1) {

B = b1;

C = c1;

}

void Get\_answer() {

cout << "Корень уравнения: " << (-1) \* (C / B) << endl;

}

void show() {

cout << "Уравнение " << B << "\*x + " << C << " = 0" << endl;

}

void Get\_answer\_podbor() {

double min = abs(0 - B \* (-20) - C);

double x = -20;

for (double i = -20; i < 20.01; i = i + 0.01) {

double temp = abs(0 - B \* (i)-C);

if (temp < min) {

min = temp;

x = i;

}

}

cout << "Приближенно, один из корней уравнения: " << x << endl;

}

};

class type6 {

private:

double A, B, C;

public:

type6(double a1, double b1, double c1) {

A = a1;

B = b1;

C = c1;

}

void Get\_answer() {

double D = B \* B - 4 \* A \* C;

if (D < 0) {

cout << "Уравнение не имеет действительных корней" << endl;

}

else {

if (D == 0) {

double x = ((-1) \* B - sqrt(D)) / (2 \* A);

cout << "Корень уравнения: " << x << endl;

}

else {

double x1 = ((-1) \* B - sqrt(D)) / (2 \* A);

double x2 = ((-1) \* B + sqrt(D)) / (2 \* A);

cout << "Корни уравнения: " << endl;

cout << "X1 = " << x1 << endl << "X2 = " << x2 << endl;

}

}

}

void show() {

cout << "Уравнение " << A << "\*x^2 + " << B << "\*x + " << C << " = 0" << endl;

}

void Get\_answer\_podbor() {

double D = B \* B - 4 \* A \* C;

if (D < 0) {

cout << "Уравнение не имеет действительных корней" << endl;

}

else {

double min = abs(0 - A \* (-20) \* (-20) - B \* (-20) - C);

double x = -20;

for (double i = -20; i < 20.01; i = i + 0.01) {

double temp = abs(0 - A \* (i) \* (i)-B \* i - C);

if (temp < min) {

min = temp;

x = i;

}

}

cout << "Приближенно, один из корней уравнения: " << x << endl;

}

}

};

class type7 {

private:

double B;

public:

type7(double b1) {

B = b1;

}

void Get\_answer() {

cout << "Корень уравнения: x = 0" << endl;

}

void show() {

cout << "Уравнение " << B << "\*x = 0" << endl;

}

};

class type8 {

private:

double A, B;

public:

type8(double a1, double b1) {

A = a1;

B = b1;

}

void Get\_answer() {

double x = ((-1) \* B) / A;

cout << "Корни уравнения: " << endl;

cout << "X1 = 0" << endl << "X2 = " << x << endl;

}

void show() {

cout << "Уравнение " << A << "\*x^2 + " << B << "\*x = 0 " << endl;

}

void Get\_answer\_podbor() {

double min = abs(0 - A \* (-20) \* (-20) - B \* (-20));

double x = -20;

for (double i = -20; i < 20.01; i = i + 0.01) {

if (i != 0) {

double temp = abs(0 - A \* i \* i - B \* (i));

if (temp < min) {

min = temp;

x = i;

}

}

}

cout << "Приближенно, один из корней уравнения: " << x << endl;

}

};

int main()

{

setlocale(LC\_ALL, "Russian");

double A, B, C;

cout << "Введите коэффициенты 1-го уравнения A, B и C через пробел: " <<

endl;

cin >> A >> B >> C;

type1 eq1;

cout << "Введите коэффициенты 2-го уравнения A, B и C через пробел: " <<

endl;

cin >> A >> B >> C;

type2 eq2(B);

cout << "Введите коэффициенты 3-го уравнения A, B и C через пробел: " <<

endl;

cin >> A >> B >> C;

type3 eq3(C);

cout << "Введите коэффициенты 4-го уравнения A, B и C через пробел: " <<

endl;

cin >> A >> B >> C;

type4 eq4(A, C);

cout << "Введите коэффициенты 5-го уравнения A, B и C через пробел: " <<

endl;

cin >> A >> B >> C;

type5 eq5(B, C);

cout << "Введите коэффициенты 6-го уравнения A, B и C через пробел: " <<

endl;

cin >> A >> B >> C;

type6 eq6(A, B, C);

cout << "Введите коэффициенты 7-го уравнения A, B и C через пробел: " <<

endl;

cin >> A >> B >> C;

type7 eq7(B);

cout << "Введите коэффициенты 8-го уравнения A, B и C через пробел: " <<

endl;

cin >> A >> B >> C;

type8 eq8(A, B);

eq1.show();

eq1.Get\_answer();

eq2.show();

eq2.Get\_answer();

eq3.show();

eq3.Get\_answer();

eq4.show();

eq4.Get\_answer();

eq4.Get\_answer\_podbor();

eq5.show();

eq5.Get\_answer();

eq5.Get\_answer\_podbor();

eq6.show();

eq6.Get\_answer();

eq6.Get\_answer\_podbor();

eq7.show();

eq7.Get\_answer();

eq8.show();

eq8.Get\_answer();

eq8.Get\_answer\_podbor();

return 0;

}