|  |  |  |
| --- | --- | --- |
| Защищено:  Гапанюк Ю.Е.    "\_\_"\_\_\_\_\_\_\_\_\_\_\_\_\_2020 г. |  | Демонстрация:  Гапанюк Ю.Е.    "\_\_"\_\_\_\_\_\_\_\_\_\_\_\_\_2020 г. |

**Отчет по лабораторной работе № 4**

**по курсу**

**Базовые компоненты интернет-технологий**

|  |  |
| --- | --- |
| ИСПОЛНИТЕЛЬ: |  |
| студент группы ИУ5Ц-52Б | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
|  | (подпись) |
| Яровенко М. В. | "\_\_"\_\_\_\_\_\_\_\_\_\_\_\_\_2020 г. |

Москва, МГТУ - 2020

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**СОДЕРЖАНИЕ ОТЧЕТА**

[1. Задание 3](#_Toc53994319)

[2. Листинг программы 3](#_Toc53994320)

[3. Результаты работы программы 5](#_Toc53994321)

# Задание

Составить программу на функциональном языке программирования для решения биквадратного уравнения с использованием алгоритма рассмотренного в разделе «Биквадратное уравнение» статьи https://ru.wikipedia.org/wiki/Уравнение\_четвёртой\_степени. Программа должна использовать алгебраические типы и механизм сопоставления с образцом.

В случае комплексных корней их вычисление не обязательно, можно выводить информацию о том, что корни комплексные.

# Листинг программы

// Яровенко Максим, ИУ5Ц-52Б

open System

type SquareRootResult =

| NoRoots

| Root of double

| TwoRoots of double \* double

| FourRoots of double \* double \* double \* double

let CalculateRoots(a:double, b:double, c:double):SquareRootResult =

let D = b\*b - 4.0\*a\*c;

if D < 0.0 then NoRoots

else if D = 0.0 then

let x21 = Math.Sqrt(-b / (2.0 \* a));

let x22 = - Math.Sqrt(-b / (2.0 \* a));

if x21 = 0.0 then Root (x21)

else TwoRoots (x21, x22)

else

let sqrtD = Math.Sqrt(D)

let x41 = Math.Sqrt((-b + Math.Sqrt(D)) / (2.0 \* a));

let x42 = -Math.Sqrt((-b + Math.Sqrt(D)) / (2.0 \* a));

let x43 = Math.Sqrt((-b - Math.Sqrt(D)) / (2.0 \* a));

let x44 = -Math.Sqrt((-b - Math.Sqrt(D)) / (2.0 \* a));

if ((-b - Math.Sqrt(D)) / (2.0 \* a)) < 0.0 then TwoRoots (x41, x42)

else if ((-b + Math.Sqrt(D)) / (2.0 \* a)) < 0.0 then TwoRoots (x43, x44)

else FourRoots (x41, x42, x43, x44)

let PrintRoots(a:double, b:double, c:double):unit =

printf "Коэффициенты: A=%A, B=%A, C=%A. " a b c

let root = CalculateRoots(a,b,c)

//Оператор сопоставления с образцом

let textResult =

match root with

| NoRoots -> "Действительных корней нет"

| Root(x21) -> "Один корень: " + x21.ToString()

| TwoRoots(x21 ,x22) -> "Два корня: " + x21.ToString() + " и " + x22.ToString()

| FourRoots(x41, x42, x43, x44) -> "Четыре корня: " + x41.ToString() + " ; " + x42.ToString() + " ; " + x43.ToString() + " ; " + x44.ToString()

printfn "%s" textResult

[<EntryPoint>]

let main argv =

printfn "%s" "Яровенко Максим, ИУ5Ц-52Б"

printfn "%s" "Данная программа решает уравнения биквадратные уравнения вида Ax^4 + Bx^2 + C = 0.\n\nТестовые примеры:"

let a4 = 1.0;

let b4 = -5.0;

let c4 = 6.0;

let a1 = 1.0;

let b1 = 0.0;

let c1 = -4.0;

let a2 = 1.0;

let b2 = 0.0;

let c2 = 0.0;

let a3 = 1.0;

let b3 = 0.0;

let c3 = 4.0;

PrintRoots(a4,b4,c4)

PrintRoots(a1,b1,c1)

PrintRoots(a2,b2,c2)

PrintRoots(a3,b3,c3)

printfn "%s" "\nВведите свои значения коэффициентов A, B, C:"

let aa = double(Console.ReadLine())

let bb = double(Console.ReadLine())

let cc = double(Console.ReadLine())

PrintRoots(aa,bb,cc)

Console.ReadLine() |> ignore

0

# Результаты работы программы

