Выполнил студент группы 348001:  
Паращенко Никита Васильевич

**Условие**  
Реализовать вычисление чисел Фиббоначи нативным и «правильным» алгоритмом. Замерить время выполнения каждой реализации алгоритмов для N = 100, 250, 500, 1000, 5000. Сравнить результаты замеров

**Решение (на Delphi)**

**Нативный алгоритм:**

program FibRecursive;

uses

System.SysUtils, System.Diagnostics;

function FibonacciRecursive(N: Integer): Int64;

begin

if N <= 1 then

Result := N

else

Result := FibonacciRecursive(N - 1) + FibonacciRecursive(N - 2);

end;

procedure vremya;

const

TestValues: array[0..4] of Integer = (10, 20, 30, 35, 40);

var

I: Integer;

Stopwatch: TStopwatch;

TimeRec: Int64;

begin

Writeln('N', #9, 'Recursive (ms)');

for I in TestValues do

begin

Writeln('Testing N = ', I);

Stopwatch := TStopwatch.StartNew;

FibonacciRecursive(I);

Stopwatch.Stop;

TimeRec := Stopwatch.ElapsedMilliseconds;

Writeln(I, #9, TimeRec);

end;

end;

begin

try

vremya;

except

on E: Exception do

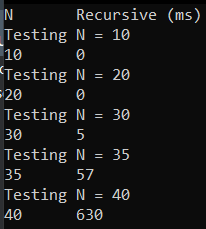
Writeln('Error: ', E.Message);

end;

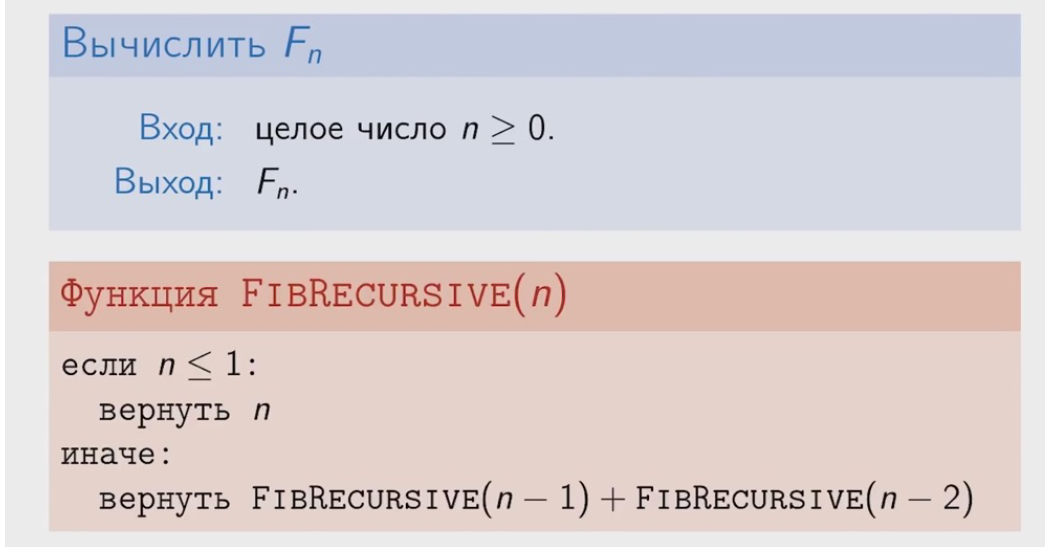
Readln;

end.

**Время выполнения:**



Для рекурсивного метода не имеет смысла брать N>40 , т.к. время выполнения будет очень большим.



**«Правильный алгоритм»:**

program FibonacciArr;

uses

System.SysUtils, System.Diagnostics;

function FibonacciArray(N: Integer): Extended;

var

Fib1, Fib2, Temp: Extended;

I: Integer;

begin

if N <= 1 then

Exit(N);

Fib1 := 0;

Fib2 := 1;

for I := 2 to N do

begin

Temp := Fib1 + Fib2;

Fib1 := Fib2;

Fib2 := Temp;

end;

Result := Fib2;

end;

procedure vremya;

const

TestValues: array[0..4] of Integer = (100, 250, 500, 1000, 5000);

RepeatCount = 10000; // Количество повторов для усреднения

var

I, J: Integer;

Stopwatch: TStopwatch;

TotalTime, TimeArr: Double;

begin

Writeln('N', #9, 'Array (ms)');

for I in TestValues do

begin

Writeln('Testing N = ', I);

Stopwatch := TStopwatch.StartNew;

TotalTime := 0;

for J := 1 to RepeatCount do

FibonacciArray(I);

Stopwatch.Stop;

TotalTime := Stopwatch.ElapsedTicks / RepeatCount;

TimeArr := TotalTime \* 1000 / Stopwatch.Frequency; // Перевод в мс

Writeln(I, #9, FormatFloat('0.000', TimeArr)); // Вывод в миллисекундах

end;

end;

begin

try

vremya;

except

on E: Exception do

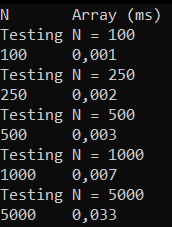
Writeln('Error: ', E.Message);

end;

Readln;

end.

**Время выполнения:**

****

Как мы видим, время выполнения программы данным методом намного быстрее,чем рекурсивный

