

Национальный исследовательский университет “Высшая Школа Экономики”,  
Факультет компьютерных наук  
Департамент программной инженерии

**«Программа, вычисляющая значение тангенса»**

Пояснительная записка к разработке консольного приложения

Исполнитель:  
Студент группы БПИ199  
Крылов Павел Юрьевич

## **Оглавление**

<b>1. Текст задания .....</b>	<b>3</b>
<b>2. Применяемые расчетные методы .....</b>	<b>4</b>
2.1 Алгоритм вычисления .....	4
2.2 Считывание числа x .....	5
2.3. Вывод результата в консоль .....	5
<b>3. Тестовые примеры .....</b>	<b>6</b>
<b>4. Список используемых источников.....</b>	<b>8</b>
<b>5. Текст программы.....</b>	<b>9</b>

## 1. Текст задания

Формулировка задания: Разработать программу, вычисляющую с помощью степенного ряда с точностью не хуже 0,05% значение функции  $\tan(x)$  для заданного параметра  $x$  (использовать FPU) [1].

## 2. Применяемые расчетные методы

### 2.1 Алгоритм вычисления

Здесь могло использоваться разложение тангенса в степенной ряд (см. рисунок 1), но проще посчитать синус и косинус с помощью степенного ряда (см. рисунок 2), косинус с помощью степенного ряда (см. рисунок 3) и вычислить тангенс как синус, деленный на косинус.

$$\begin{aligned}\tan x &= \sum_{n=0}^{\infty} \frac{U_{2n+1} x^{2n+1}}{(2n+1)!} \\ &= \sum_{n=1}^{\infty} \frac{(-1)^{n-1} 2^{2n} (2^{2n} - 1) B_{2n} x^{2n-1}}{(2n)!} \\ &= x + \frac{1}{3} x^3 + \frac{2}{15} x^5 + \frac{17}{315} x^7 + \dots, \quad \text{for } |x| < \frac{\pi}{2}.\end{aligned}$$

Рисунок 1. Разложение тангенса в степенной ряд

$$\begin{aligned}\sin x &= x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \dots \\ &= \sum_{n=0}^{\infty} \frac{(-1)^n x^{2n+1}}{(2n+1)!}\end{aligned}$$

Рисунок 2. Разложение синуса в степенной ряд

$$\begin{aligned}\cos x &= 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} + \dots \\ &= \sum_{n=0}^{\infty} \frac{(-1)^n x^{2n}}{(2n)!}.\end{aligned}$$

Рисунок 3. Разложение косинуса в степенной ряд

## **2.2 Считывание числа x**

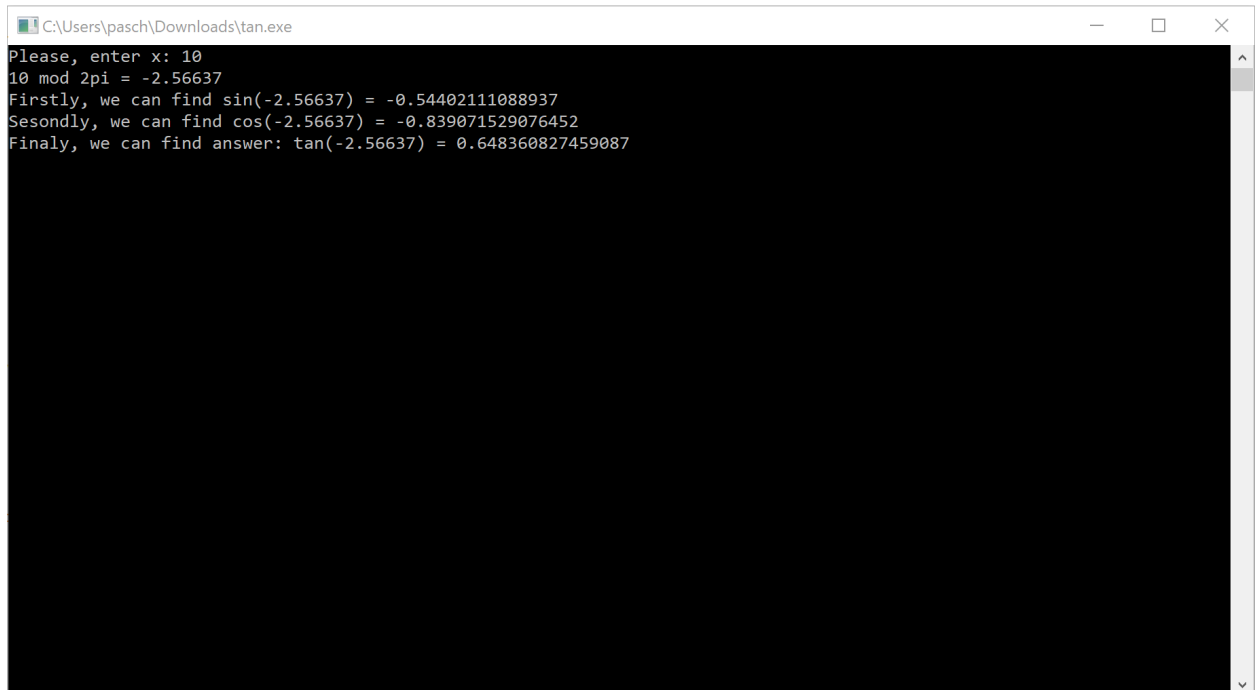
Число x считывается с командной строки либо из первого аргумента argv. Для того чтобы функция scanf работала с 64-битными числами с плавающей точкой, используется "%lf" вместо "%f".

## **2.3. Вывод результата в консоль**

Для вывода используется функция printf и "%23.20g" вместо "%f", из-за того что "%f" для больших чисел выводит в консоль большое количество лишних нулей.

### 3. Тестовые примеры

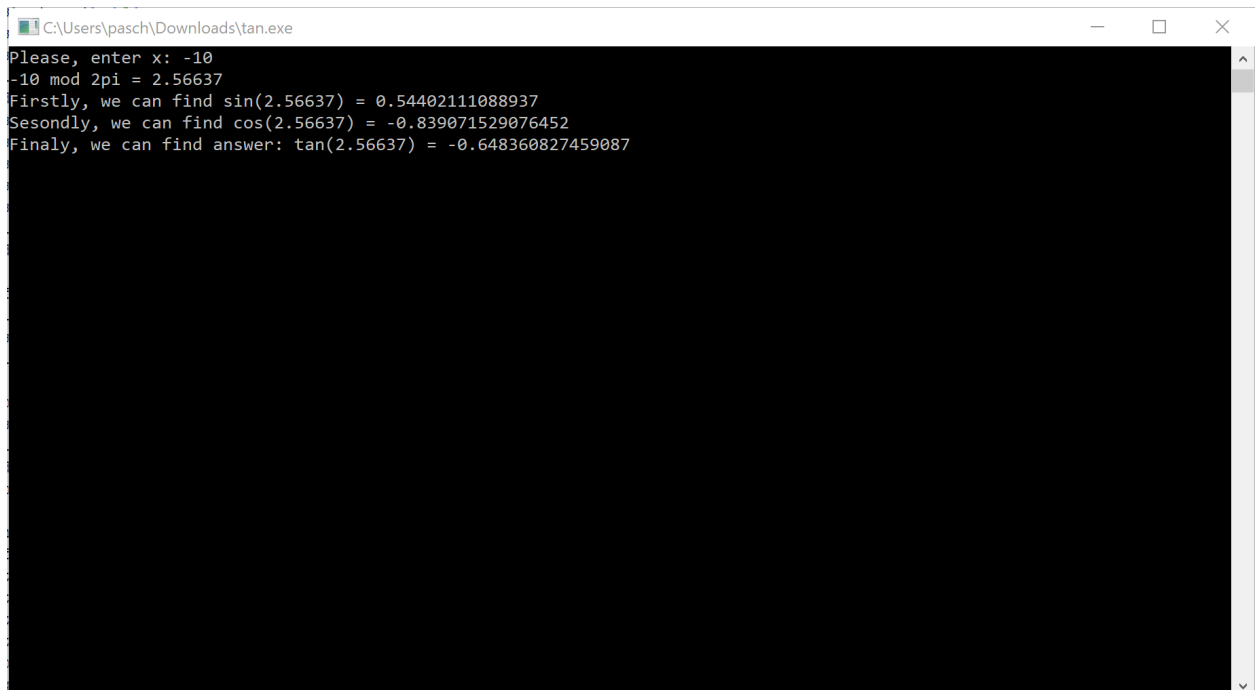
Программа корректно работает для положительных чисел (см. рисунок 4).



```
C:\Users\pasch\Downloads\tan.exe
Please, enter x: 10
10 mod 2pi = -2.56637
Firstly, we can find sin(-2.56637) = -0.54402111088937
Sesondly, we can find cos(-2.56637) = -0.839071529076452
Finally, we can find answer: tan(-2.56637) = 0.648360827459087
```

Рисунок 4. Корректные входные данные, положительное число

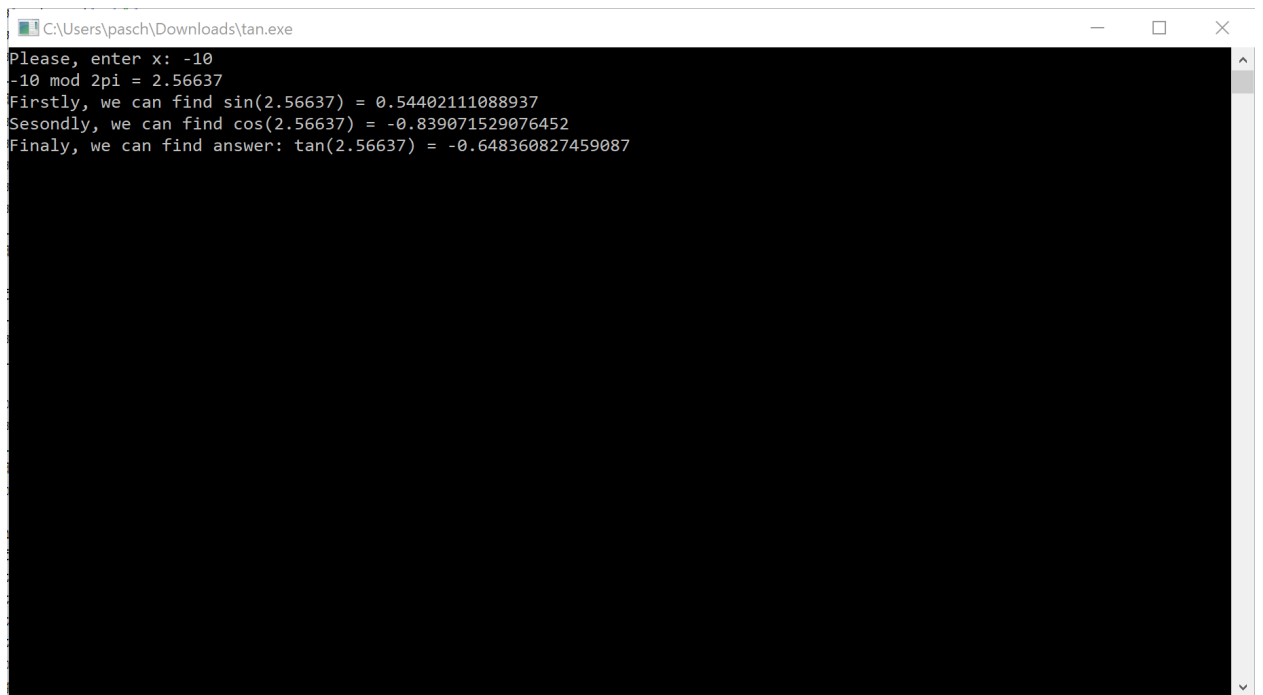
Программа корректно работает для отрицательных чисел (см. рисунок 5).



```
C:\Users\pasch\Downloads\tan.exe
Please, enter x: -10
-10 mod 2pi = 2.56637
Firstly, we can find sin(2.56637) = 0.54402111088937
Sesondly, we can find cos(2.56637) = -0.839071529076452
Finally, we can find answer: tan(2.56637) = -0.648360827459087
```

Рисунок 5. Корректные входные данные, отрицательное число

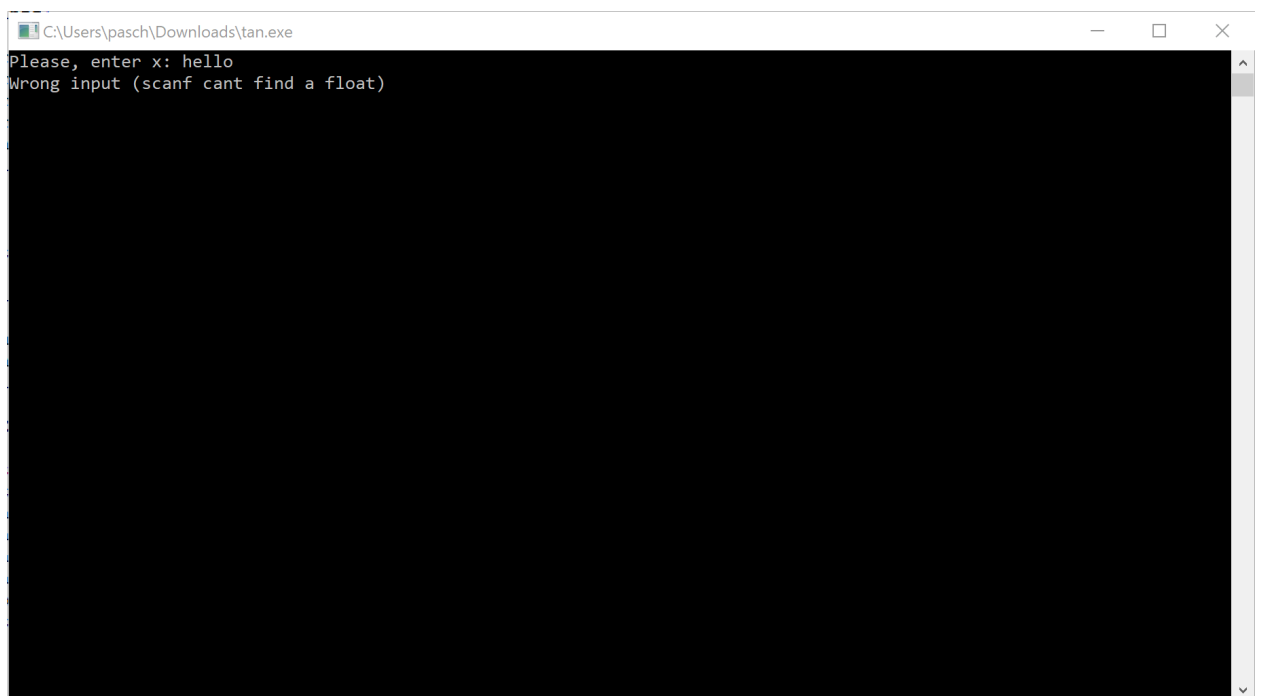
Программа корректно работает для нуля (см. рисунок 6).



```
C:\Users\pasch\Downloads\tan.exe
Please, enter x: -10
-10 mod 2pi = 2.56637
Firstly, we can find sin(2.56637) = 0.54402111088937
Secondly, we can find cos(2.56637) = -0.839071529076452
Finally, we can find answer: tan(2.56637) = -0.648360827459087
```

Рисунок 6. Корректные входные данные, ноль

Программа выводит сообщение об ошибке при попытке ввода строки символов (см. рисунок 7).



```
C:\Users\pasch\Downloads\tan.exe
Please, enter x: hello
Wrong input (scanf cant find a float)
```

Рисунок 7. Некорректные входные данные, строка символов

#### **4. Список используемых источников**

[1] Программирование на языке ассемблера. Микропроект. Требования к оформлению. 2020-2021 уч.г. [Электронный ресурс]. // URL: <http://softcraft.ru/edu/comparch/tasks/mp01/> (Дата обращения: 31.10.2020, режим доступа: свободный)

[2] FLAT ASSEMBLER 1.64 — МАНУАЛ ПРОГРАММЕРА [Электронный ресурс]. // URL: <http://flatassembler.narod.ru/fasm.htm#2-1-13> (Дата обращения: 31.10.2020, режим доступа: свободный)

[3] Ряд Тейлора — Википедия [Электронный ресурс]. // URL: [https://ru.wikipedia.org/wiki/Ряд\\_Тейлора](https://ru.wikipedia.org/wiki/Ряд_Тейлора) (Дата обращения: 31.10.2020, режим доступа: свободный)



## 5. Текст программы

```
format PE console
entry start
include 'win32a.inc'

section '.code' code readable executable
sin:
    finit
    fld1
    fstp QWORD[delta]

    mov ecx, 0
sin_loop:
    inc ecx
    ; delta *= x / i;
    fld QWORD[x]
    mov [i], ecx
    fidiv dword[i]
    fmul QWORD[delta]
    fstp QWORD[delta]

    ; i % 2 == 0
    mov eax, ecx
    and eax, 1
    cmp eax, 0
    je sin_loop

    ; res += delta;
    fld QWORD[delta]
    fadd QWORD[sinres]
    fstp QWORD[sinres]

    ; delta *= -1;
    fld QWORD[delta]
    fchs
    fstp QWORD[delta]

    cmp ecx, 1000
    jl sin_loop
sin_end:
    ret

cos:
    finit
    fld1
    ; res = 1.0;
    fst QWORD[cosres]
    fchs
    fstp QWORD[delta]
```

```

mov ecx, 0
cos_loop:
inc ecx
; delta *= x / i;
fld QWORD[x]
mov [i], ecx
fidiv dword[i]
fmul QWORD[delta]
fstp QWORD[delta]

; i % 2 != 0
mov eax, ecx
and eax, 1
cmp eax, 0
jne cos_loop

; res += delta;
fld QWORD[delta]
fadd QWORD[cosres]
fstp QWORD[cosres]

; delta *= -1;
fld QWORD[delta]
fchs
fstp QWORD[delta]

cmp ecx, 1000
jl cos_loop
cos_end:
ret

capX:
finit
fldpi
fimul dword[two]
fld QWORD[x]
fst QWORD[oldx]
fprem1
fstp QWORD[x]

fld QWORD[oldx]
fld QWORD[x]
fcompp
fstsw ax
sahf
je capX_ret

push dword[x+4]
push dword[x]
push dword[oldx+4]
push dword[oldx]

```

```

    push printfFormat_mod
    call [printf]
    add esp, 20
capX_ret:
    ret

tan:
    finit
    fld QWORD[sinres]
    fdiv QWORD[cosres]
    fstp QWORD[tanres]
    ret

start:
    push askForX
    call [printf]
    add esp, 4
    push x
    push scanfFormat
    call [scanf]
    add esp, 8
    cmp eax, 1
    jne wrongInput

    call capX
    call sin
    call cos
    call tan

    push dword[sinres+4]
    push dword[sinres]
    push dword[x+4]
    push dword[x]
    push printfFormat_sin
    call [printf]
    add esp, 20
    push dword[cosres+4]
    push dword[cosres]
    push dword[x+4]
    push dword[x]
    push printfFormat_cos
    call [printf]
    add esp, 20
    push dword[tanres+4]
    push dword[tanres]
    push dword[x+4]
    push dword[x]
    push printfFormat_tan
    call [printf]
    add esp, 20

```

```

safeExit:
call [getch]
push 0
call [exit]

wrongInput:
push wrongFromat
call [printf]
add esp, 4
jmp safeExit

```

```

section '.data' data readable writable

```

```

scanfFormat: db '%lf',0
printfFormat_mod: db '%g mod 2pi = %g',10,0
printfFormat_sin: db 'Firstly, we can find sin(%g) = %.15g',10,0
printfFormat_cos: db 'Sesondly, we can find cos(%g) = %.15g',10,0
printfFormat_tan: db 'Finaly, we can find answer: tan(%g) = %.15g',10,0
askForX: db 'Please, enter x: ',0
wrongFromat: db 'Wrong input (scanf cant find a float)',10,0
two: dd 2
i: dd 0
x: dq 0
oldx: dq 0
sinres: dq 0
cosres: dq 0
tanres: dq 0
delta: dq 0

```

```

section '.idata' import code readable

```

```

library msvcrt, 'msvcrt.dll'
import msvcrt, printf, 'printf', scanf, 'scanf', exit, '_exit', getch,
'_getch'

```