

**НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

Факультет компьютерных наук
Департамент программной инженерии
Дисциплина: «Архитектура вычислительных систем»

ВЫЧИСЛЕНИЕ ЗНАЧЕНИЯ ФУНКЦИИ $\text{ARCTAN}(X)$

Пояснительная записка

Выполнил:
Снимщиков Илья,
студент гр. БПИ197.

Москва
2020

Содержание

1. Текст задания	2
2. Применяемые расчетные методы.....	3
2.1. Теория решения задания.....	3
3. Тестирование программы.....	4
3.1. Корректные значения.....	4
3.2. Некорректные значения	5
ПРИЛОЖЕНИЕ 1.....	6
Список литературы	6
ПРИЛОЖЕНИЕ 2.....	7
Код программы	7

1. Текст задания

Разработать программу, вычисляющую с помощью степенного ряда с точностью не хуже 0,05% значение функции $\arctan(x)$ для заданного параметра x (использовать FPU)

2. Применяемые расчетные методы

2.1. Теория решения задания

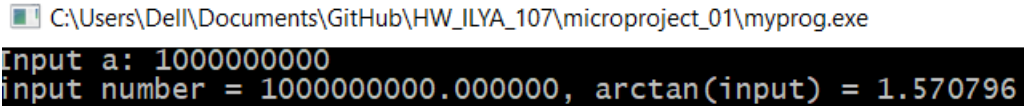
$$\operatorname{arctg} \alpha = \alpha - \frac{\alpha^3}{3} + \frac{\alpha^5}{5} - \frac{\alpha^7}{7} + \dots + (-1)^n \cdot \frac{\alpha^{2n+1}}{2n+1} + \dots = \sum_{n=0}^{\infty} \frac{(-1)^n \alpha^{2n+1}}{2n+1}$$

Область сходимости ряда: $-1 \leq \alpha \leq 1$ или, то же самое: $|\alpha| \leq 1$

Если $|\alpha| > 1$, то $\operatorname{arctg}(\alpha) = \operatorname{arcctg}(1/\alpha) = \pi - \operatorname{arctg}(1/\alpha)$

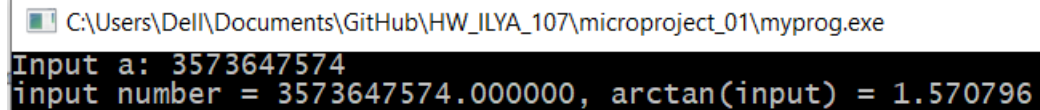
3. Тестирование программы

3.1. Корректные значения



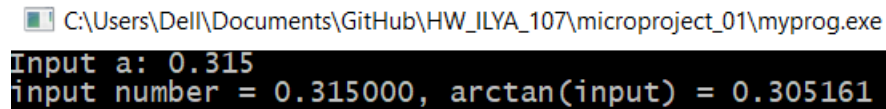
```
C:\Users\Dell\Documents\GitHub\HW_ILYA_107\microproject_01\myprog.exe
Input a: 1000000000
input number = 1000000000.000000, arctan(input) = 1.570796
```

1. Рисунок 1. Вычисление арктангенса большого положительного числа



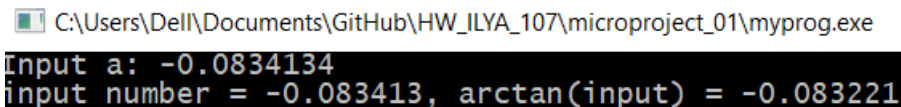
```
C:\Users\Dell\Documents\GitHub\HW_ILYA_107\microproject_01\myprog.exe
Input a: 3573647574
input number = 3573647574.000000, arctan(input) = 1.570796
```

2. Рисунок 2. Вычисление арктангенса большого отрицательного числа



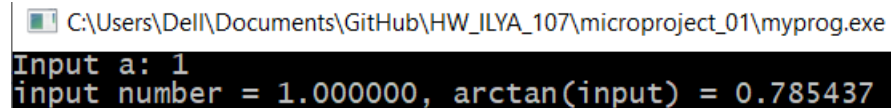
```
C:\Users\Dell\Documents\GitHub\HW_ILYA_107\microproject_01\myprog.exe
Input a: 0.315
input number = 0.315000, arctan(input) = 0.305161
```

3. Рисунок 3. Вычисление арктангенса маленького положительного числа



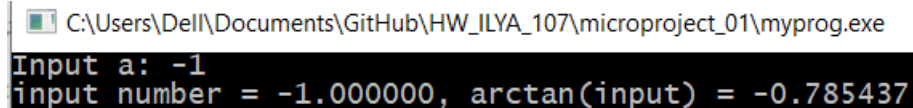
```
C:\Users\Dell\Documents\GitHub\HW_ILYA_107\microproject_01\myprog.exe
Input a: -0.0834134
input number = -0.083413, arctan(input) = -0.083221
```

4. Рисунок 4. Вычисление арктангенса маленького отрицательного числа



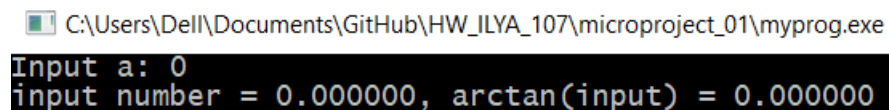
```
C:\Users\Dell\Documents\GitHub\HW_ILYA_107\microproject_01\myprog.exe
Input a: 1
input number = 1.000000, arctan(input) = 0.785437
```

5. Рисунок 5. Вычисление арктангенса единицы



```
C:\Users\Dell\Documents\GitHub\HW_ILYA_107\microproject_01\myprog.exe
Input a: -1
input number = -1.000000, arctan(input) = -0.785437
```

6. Рисунок 6. Вычисление арктангенса минус единицы



```
C:\Users\Dell\Documents\GitHub\HW_ILYA_107\microproject_01\myprog.exe
Input a: 0
input number = 0.000000, arctan(input) = 0.000000
```

7. Рисунок 7. Вычисление арктангенса нуля

3.2. Некорректные значения

Все разумные(меньше 10^{100} и больше -10^{100}) числовые значения корректны

Список литературы

1. Symply FPU by **Raymond Filiatreault**. [Электронный ресурс] // URL: <https://www.website.masmforum.com/tutorials/fptute/> (дата обращения: 02.11.2020)
2. SoftCraft. [Электронный ресурс] // URL: <http://softcraft.ru/edu/comparch/> (дата обращения: 02.11.2020)

Код программы

```
format PE console
```

```
entry start
```

```
include 'win32a.inc'
```

```
section '.code' code readable executable
```

```
start:
```

```
    FINIT ; инициализация сопроцессора
```

```
    call Input ;ввод числа
```

```
    FLD [input] ; загрузка числа в вершину стека st(0)
```

```
    call Atan
```

```
    FSTP [result] ; запись в память результата вычисления арктангенса
```

```
    ; вывод результата
```

```
    invoke printf, fmt_string, dword[input], \
```

```
        dword[input+4], \
```

```
        dword[result],dword[result+4]
```

```
    invoke getch
```

```
    invoke ExitProcess, 0 ; выход из программы с кодом возврата 0
```

```
;Ввод числа
```

```
Input:
```

```
    push strScanFloatText
```

```
    call [printf]
```

```
    add esp, 4
```

```
    push input ;ввод числа
```

```
    push strScanFloat
```

```
    call [scanf]
```

```
    add esp, 8
```

```
ret
```


;Вычисление арктангенса числа из st(0)
 ;После вычислений число перемещается в sp(1)
 ;Результат находится в SP(0)
 ;Для вычислений используется 6 ячеек

Atan:

```
FLDZ
FCOMIP st1
jne notzero
FLDZ
ret
;проверяем, что |input| < 1
```

notzero:

```
FLD1
FCOMIP st1
jb normpos
FLD1
FCHS
FCOMIP st1
ja normneg
```

```
FLDZ; загружаем 0 в стек
jmp startloop
```

normneg:

```
;Вычитаем pi просто для красоты, чтобы результаты были от -pi/2 до pi/2
FLDPI
FCHS
jmp norm
```

normpos:

```
FLDPI
```

norm:

```
;если модуль больше 1, то нормализуем согласно алгоритму
FLD1
FCHS
FDIVRP st2, st
FLD1
FADD st, st
FDIVP st1, st
```

startloop:

```
FLD1
FLD st2
```

continueloop:

```
;st0 - текущий  $x^{2n} + 1$ 
;st1 - текущий  $2n + 1$ 
;st2 - текущий результат
;st3 - входное число
```

;Прибавление очередного слагаемого к результату

FLD st1

FDIVR st, st1

FADD st3, st

;Проверка, достигнута ли нужная точность

FDIV st, st3

FABS

FLD [eps]

FCOMIP st1

FSTP st

;Выход из алгоритма, если точность достигнута

ja endloop

;Обновление счетчиков и вспомогательных переменных для следующей итерации цикла

FCHS

FMUL st, st3

FMUL st, st3

FLD1

FADD st, st

FADDP st2, st

jmp continueloop

endloop:

;Выход из алгоритма

FSTP st

FSTP st

ret

;-----third act - including HeapApi-----

section '.data' data readable writable

fmt_string db "input number = %lf, arctan(input) = %lf", 10, 0

strScanFloat db '%lf', 0

strScanFloatText db 'Input a: ', 0

input dq ? ; входное число

eps dq 0.0001 ; допуск

result dq ? ; результат вычисления (формат двойной точности)

```
section '.idata' import data readable
    library kernel, 'kernel32.dll',\
        msvcrt, 'msvcrt.dll',\
        user32, 'USER32.DLL'

include 'api\user32.inc'
include 'api\kernel32.inc'
import kernel,\
    ExitProcess, 'ExitProcess',\
    HeapCreate, 'HeapCreate',\
    HeapAlloc, 'HeapAlloc'
include 'api\kernel32.inc'
import msvcrt,\
    printf, 'printf',\
    sprintf, 'sprintf',\
    scanf, 'scanf',\
    getch, '_getch'
```