**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

**Выполнила**: Асташкина Мария Николаевна   
**Группа**: БПИ 196  
**Вариант: 2**

**ЗАДАНИЕ**

Разработать программу, вычисляющую с помощью степенного ряда с точностью не хуже 0,1% значение функции √1+x для заданного параметра x (использовать FPU).

**ИСХОДНЫЙ КОД**

format PE console

include 'win32a.inc'

entry start

section 'data' data readable writeable

string db 'Enter number: ',0 ;строки для вывода

string2 db 'Number must be from -1 to 1',0 ;строки для вывода

string3 db 'Sqrt(%f+1)=%f ',0

scanf\_float db '%f', 0 ;строка для ввода float

number dd ?;переменная для вводимого числа

out1 dq ? ;переменные для вывода

out2 dq ?

accuracy dd 0.001;точность

F1 dd 0.5 ;константы

F2 dd 2.0

n dd ?;переменная цикла

section 'text' code executable readable

start:

push string

call [printf]

push number;получение числа

push scanf\_float

call [scanf]

;область определения функции [-1,1]

finit ;начало работы с FPU, проверка переменной на значение не больше еденицы

fld [number];загрудаем число

fld1;загружем еденицу

fcomip st1;сравниваем числа

jae .ok1;если число больше

push string2;выводим сообщение об ошибке

call [printf]

jmp .g;из завершаем работу программы

.ok1:

fld1;проверяем число на значение не меньше -1

fchs

fcomip st1

jbe .ok2;если оно не меньше пропускаем вывод ошибки

push string2

call [printf];вывод ошибки и выход из программы

jmp .g

.ok2:

;float func(float x){

;float g = x/2;

;float result = 1.0;

;int i = 1;

;while (fabs(g) > 0.001){

; result += g;

; g \*= x \* ((0.5 - i)) / (i + 1);

; i++;

;}

;return result;

;}

finit;для начала работы с FPU

fld [accuracy];загружаем точность

fld [number];получаем начальное значение для g, загружаем x

fdiv [F2];делим его на 2

fld1;загружаем 1 для переменной result

mov [n],1;ставим счетчик цикла на 1

.loop:;начало цикла

fadd st0,st1;result += g;

fld [number];загружаем x

fld [F1];загружаем 0.5f

fisub [n];вычитаем из него i (0.5-i)

fmulp st1,st0;умножаем на x (x\*(0.5-i))

fild [n];еще раз загружаем i

fld1;загружаем 1

faddp st1,st0;прибавляем, получаем i+1

fdivp st1,st0;делим получившиеся числа, получаем x\*(0.5-i)/(i+1)

fmul st0,st2;умножаем на g

fstp st2;результат записываем обратно в g

inc [n];увеличиваем счетчик цикла

fld st1;получаем g

fabs;получаем его модуль

fcomip st3;сравниваем его с точностью

jae .loop;если g больше точности, начинаем заново

fstp qword[out2];выносим ответ в переменную

fld [number];расширяем number до double

fstp qword[out1]

push dword[out2+4];вывод

push dword[out2]

push dword[out1+4]

push dword[out1]

push string3

call [printf]

.g:

call [getch];ждем

push 0

call [ExitProcess];завершение программы

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'

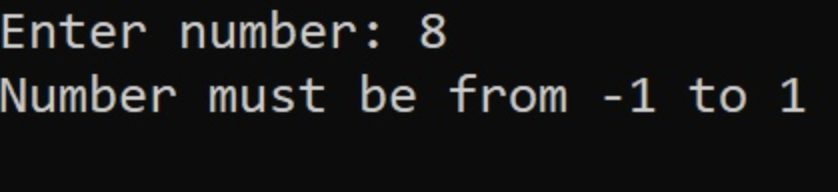
import msvcrt,\

printf, 'printf',\

scanf, 'scanf',\

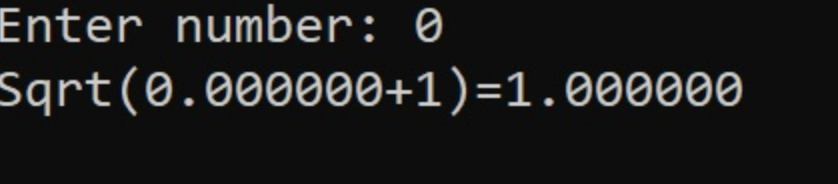
getch, '\_getch'

**ТЕСТЫ**

1. 
2. **Изображение выглядит как текст

   Автоматически созданное описание**
3. **Изображение выглядит как текст

   Автоматически созданное описание**
4. **Изображение выглядит как текст

   Автоматически созданное описание**
5. ****