

.286C

yulya segment para 'code'

assume cs:yulya, ds:yulya,es:yulya,ss:yulya

org 100h

begin: jmp main

X dd -2.0

TMP dd ?

FOUR dd 4.0

FIVE dd 5.0

main proc near

;Y = arctg ((1+x)/(1-x))/4

FINIT

FLD1 ;ST(0)=1

FLD X ;ST(0)=X ST(1)=1

FSUB ;ST(0)=1-X

FSTP TMP ;TMP=1-X ST(0)=0

FLD1 ;ST(0)=1

FLD X ;ST(0)=X ST(1)=1

FADD ;ST(0)=1+X

FLD TMP ;ST(0)=1-X ST(1)=1+X

FPATAN ;ST(0)=arctg((1+x)/(1-x))

FLD FOUR ;ST(0)=4 ST(1)=arctg((1+x)/(1-x))

FDIV ;ST(0)=arctg((1+x)/(1-x))/4

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Вывод\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

call outfloat

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ret

main endp

outfloat proc near

push ax

push cx

push dx

; Формируем кадр стэка, чтобы хранить в нём десятку

; и ещё какую-нибудь цифру.

push bp

mov bp, sp

push 10

push 0

; Проверяем число на знак, и если оно отрицательное,

ftst

fstsw ax

sahf

jnc @of1

; то выводим минус

mov ah, 02h

mov dl, '-'

int 21h

; и оставляем модуль числа.

fchs

; Пояснение далее пойдёт на примере. ; ST(0) ST(1) ST(2) ST(3) ...

; Отделим целую часть от дробной. ; 73.25 ... что-то не наше

@of1: fld1 ; 1 73.25 ...

fld st(1) ; 73.25 1 73.25 ...

; Остаток от деления на единицу даст дробную часть.

fprem ; 0.25 1 73.25 ...

; Если вычесть её из исходного числа, получится целая часть.

fsub st(2), st ; 0.25 1 73 ...

fxch st(2) ; 73 1 0.25 ...

; Сначала поработаем с целой частью. Считать количество цифр будем в CX.

xor cx, cx

; Поделим целую часть на десять,

@of2: fidiv word ptr [bp - 2] ; 7.3 1 0.25 ...

fxch st(1) ; 1 7.3 0.25 ...

fld st(1) ; 7.3 1 7.3 0.25 ...

; отделим дробную часть - очередную справа цифру целой части исходного числа,-

fprem ; 0.3 1 7.3 0.25 ...

; от чатсного оставим только целую часть

fsub st(2), st ; 0.3 1 7 0.25 ...

; и сохраним цифру

fimul word ptr [bp - 2] ; 3 1 7 0.25 ...

fistp word ptr [bp - 4] ; 1 7 0.25 ...

inc cx

; в стэке.

push word ptr [bp - 4]

fxch st(1) ; 7 1 0.25 ...

; Так будем повторять, пока от целой части не останется ноль.

ftst

fstsw ax

sahf

jnz short @of2

; Теперь выведем её.

mov ah, 02h

@of3: pop dx

; Вытаскиваем очередную цифру, переводим её в символ и выводим.

add dl, 30h

int 21h

; И так, пока не выведем все цифры.

loop @of3 ; 0 1 0.25 ...

; Итак, теперь возьмёмся за дробную часть, для начала проверив её существование.

fstp st(0) ; 1 0.25 ...

fxch st(1) ; 0.25 1 ...

ftst

fstsw ax

sahf

jz short @of5

; Если она всё-таки ненулевая, выведем точку

mov ah, 02h

mov dl, '.'

int 21h

; и не более шести цифр дробной части.

mov cx, 6

; Помножим дрообную часть на десять,

@of4: fimul word ptr [bp - 2] ; 2.5 1 ...

fxch st(1) ; 1 2.5 ...

fld st(1) ; 2.5 1 2.5 ...

; отделим целую часть - очередную слева цифру дробной части исходного числа,-

fprem ; 0.5 1 2.5 ...

; оставим от произведения лишь дробную часть,

fsub st(2), st ; 0.5 1 2 ...

fxch st(2) ; 2 1 0.5 ...

; сохраним полученную цифру во временной ячейке

fistp word ptr [bp - 4] ; 1 0.5 ...

; и сразу выведем.

mov ah, 02h

mov dl, [bp - 4]

add dl, 30h

int 21h

; Теперь, если остаток дробной части ненулевой

fxch st(1) ; 0.5 1 ...

ftst

fstsw ax

sahf

; и мы вывели менее шести цифр, продолжим.

loopnz @of4 ; 0 1 ...

; Итак, число выведено. Осталось убрать мусор из стэка.

@of5: fstp st(0) ; 1 ...

fstp st(0) ; ...

; Точнее, стэков.

leave

pop dx

pop cx

pop ax

ret

outfloat endp

yulya ends

end begin

