**Правительство Российской Федерации**

Федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский университет «Высшая школа экономики»

Факультет компьютерных наук

**Пояснительная записка к микропроекту**

**По дисциплине**

**“Архитектура вычислительных систем”**

**На тему**

**“Определение число элементов последовательности Падована от 1 до беззнакового двойного машинного слова”**

Работу выполнил

Студент группы БПИ-194 \_\_\_\_\_\_\_\_\_\_\_\_\_ И.С. Попов

подпись, дата

Работу проверил \_\_\_\_\_\_\_\_\_\_\_\_\_ А.И. Легалов

подпись, дата

Москва 2020

Содержание

[Постановка задачи 3](#_Toc55129079)

[Описание алгоритма 3](#_Toc55129080)

[Пример работы программы 3](#_Toc55129081)

[Приложение 4](#_Toc55129082)

[Список использованной литературы 6](#_Toc55129083)

# Постановка задачи

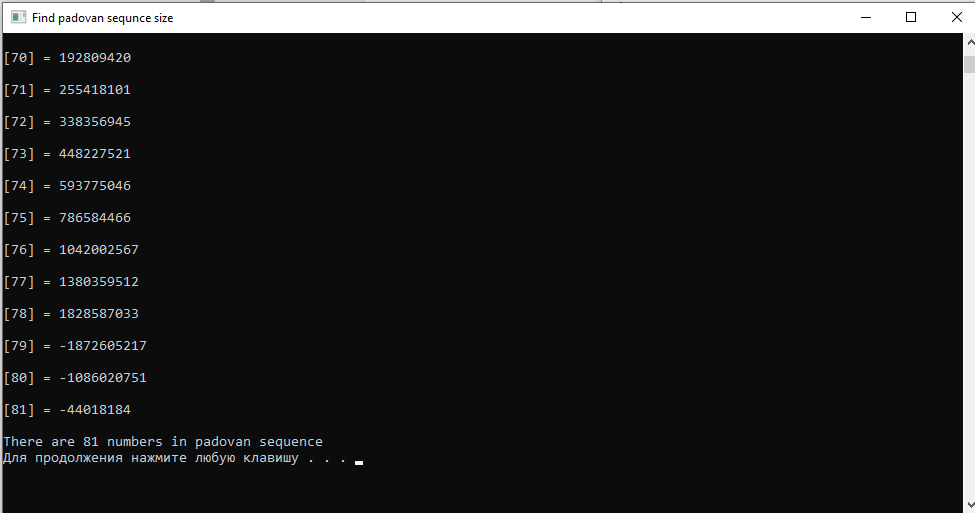
Разработать программу, определяющую число элементов последовательности Падована от 1 до беззнакового двойного машинного слова.

# Описание алгоритма

Алгоритм реализован процелурой Padovan.

В регистр eax помещается новый элемент(сумма i-2 и i-3). Если произошло целочисленное переполнение, то выходим из цикла. Иначе с использование регистров сдвигаем старые элементы на 1 назад(value - prev\_value, prev\_value – >prev\_prev\_value) а новый элемент помещается в value. Далее возвращаемся в начало цикла(jmp).

# Пример работы программы

Так как входных данных в задаче нет, в качестве теста был произведен вывод всех элементов последовательности до нужного номера

Так как при выводе через printf используется знаковое представление числа, то выведены отрицательные числа, но программа завершается(выходит из цикла) при беззнаковом переполнении.

# Приложение

Код программы:

format PE console

entry start

;------------------------------------------

include "win32a.inc"

;-------------------------------------------

section '.rdata' data readable writeable

title db "Find padovan sequnce size",0

answer db "There are %d numbers in padovan sequence",10,0

\_pause db "pause",0

prev\_prev\_value dd 1

prev\_value dd 1

value dd 1

count dd 3 ; первые 3 элемента(единицы) заданы сразу

;-----------------------------------------

section '.text' code executable

start:

invoke SetConsoleTitle, title

call Padowan

invoke ExitProcess, 0

;------------------------------------------

proc Padowan

for\_loop:

xor eax, eax

xor ebx, ebx

mov eax, [prev\_value]

add eax, [prev\_prev\_value]

; Выход при переполнении(беззнаковом)

jc for\_end

mov ebx, [value]

mov [value], eax

mov eax, [prev\_value]

mov [prev\_value], ebx

mov [prev\_prev\_value], eax

inc [count]

jmp for\_loop

for\_end:

invoke printf, answer, [count]

invoke system, \_pause

ret

endp

;-----------------------------

section '.idata' data readable import

library kernel32, 'kernel32.dll',\

msvcrt, 'msvcrt.dll'

import kernel32,\

ExitProcess, 'ExitProcess',\

SetConsoleTitle, 'SetConsoleTitleA'

import msvcrt,\

printf, 'printf',\

system, 'system',\

scanf, 'scanf'

# Список использованной литературы

1. Программирование на языке ассемблера http://natalia.appmat.ru/c&c++/assembler.html