

**Правительство Российской Федерации**  
**Федеральное государственное автономное образовательное**  
**учреждение высшего образования «Национальный исследовательский**  
**университет «Высшая школа экономики»**  
Факультет компьютерных наук  
Департамент программной инженерии

**Пояснительная записка к домашнему заданию**  
**По дисциплине**  
**«Архитектура вычислительных систем»**

Работу выполнил:  
Студент группы БПИ-194, Аникеев Егор Васильевич  
Преподаватель:  
Легалов Александр Иванович

**Москва 2020**

## Задание

Разработать программу, определяющую наименьшее число с плавающей точкой методом дихотомии и сравнения с единицей.

## Решение

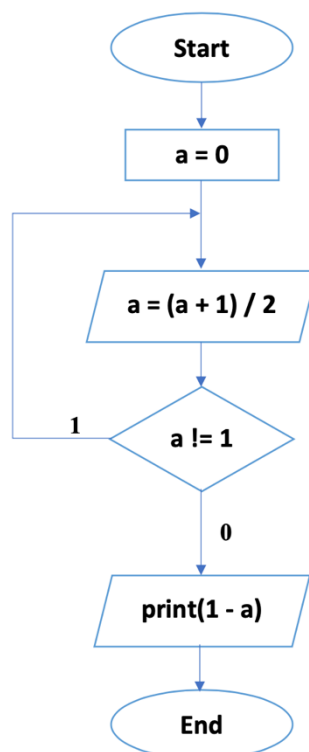
Метод дихотомии или метод бисекции - метод, при которым данный отрезок делится пополам, пока не получим результат с заданной точностью.

Адаптируем данный метод под задачу. Возьмем отрезок  $[0; 1)$  и будем смещать левую границу, пока не получим максимальную точность, которой будем добиваться сравнением с единицей. Смещение границы будем осуществлять так: левая граница = (левая граница + правая граница) / 2. Например:

- 1 итерация:  $[0; 1) \rightarrow [0,5; 1)$
- 2 итерация:  $[0,5; 1) \rightarrow [0,75; 1)$
- и так далее

В итоге мы должны получить число вроде 0,99999999..., в зависимости от того, сколько знаков после запятой поддерживает тип.

Для воспроизведения алгоритма нарисует блок-схему, которая будет реализована на ассемблере.



## Текст программы

```
format PE

entry main

include "win32ax.inc"

section '.data' data readable writeable
    two dq 2.0 ;делитель
    result dq ? ; результирующая переменная
    one dq 1.0
    print dq ? ; переменная для печати
    output db 256 dup(?) ;контейнер строки вывода

section '.code' code executable

main:

    finit

    fld qword[one] ; добавить в стек
    fstp qword [result] ; сохранить head стека в result и убрать
    find_min_loop:
        fld qword [two]
        fld qword [one]
        fld qword [result]
        ; стек имеет вид s0 - result, s1 - one, s2 - two
        fdiv st0, st2 ; делим на 2, по методу дихотомии
        fcomi st1 ; если переменная переполнилась и стала равна 1
        je done
        ;вывод промежуточного результата
        fst qword [print]
        invoke sprintf, output, ' on iter %.16f',dword[print],
dword[print+4]
        invoke MessageBox, 0, output, "Min loop", MB_OK
        fadd st0, st1 ;добавить к вершине стека(левая граница) 1
        fst qword [result] ; поместить вершину стека в result
        jmp find_min_loop ; начать цикл заново

    done:
        ; вывести результат
        invoke sprintf, output, ' is %.16f',dword[print], dword[print+4]
        invoke MessageBox, 0, output, "Min loop", MB_OK

    invoke ExitProcess, 0

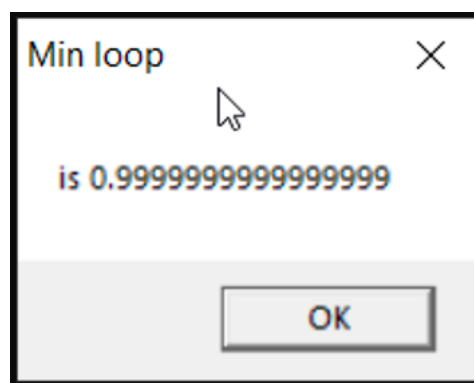
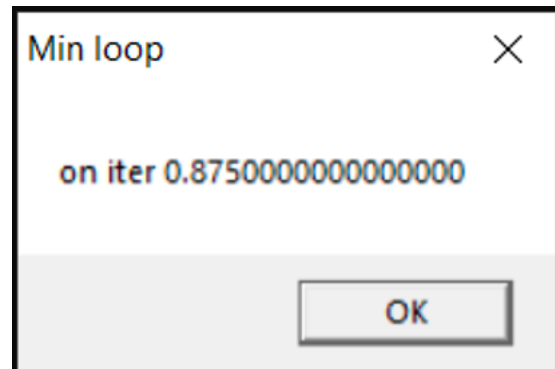
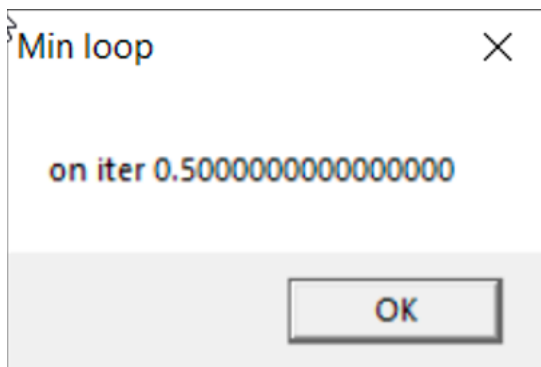
section '.idata' import data readable
    library kernel, 'kernel32.dll',\
        msvcrt, 'msvcrt.dll',\
        user32, 'USER32.DLL'

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', \
    sprintf, 'sprintf'
```

### Изображения тестов

На изображениях 1 и 2 показаны промежуточные значения результата, на рисунке 3 конечно полученное значение.



## Список используемых источников

1. Метод дихотомии  
([http://www.machinelearning.ru/wiki/index.php?title=Методы\\_дихотомии](http://www.machinelearning.ru/wiki/index.php?title=Методы_дихотомии))
2. Легалов А.И.(2020) «Разработка программ на ассемблере. Использование сопроцессора с плавающей точкой»  
(<http://softcraft.ru/edu/comparch/practice/asm86/05-fpu/>)
3. YouTube “Яша добрый хакер” (2018) «Канал Яша добрый хакер»  
(<https://www.youtube.com/user/yashechka85/videos>)