

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

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

**Микропроект по дисциплине
«Архитектура вычислительных систем»**

На тему:

«Программа вычисления корня квадратного по итерационной формуле
Герона Александрийского с точностью не хуже 0,05%»

Пояснительная записка

Выполнил:
Моторкин Владимир,
студент гр. БПИ198.

**Москва
2020**

Содержание

1. Текст задания.....	2
2. Применяемые расчетные методы.....	2
2.1. Теория решения задания.....	2
2.2. Описание переменных программы и макросов	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

Вариант 21.

Разработать программу вычисления корня квадратного по итерационной формуле Герона Александрийского с точностью не хуже 0,05% (использовать FPU).

2. Применяемые расчетные методы

2.1. Теория решения задания

В данном задании необходимо найти квадратный корень при помощи итерационной формулы Герона Александрийского.

Итерационная формула Герона имеет вид:

$$x_{n+1} = \frac{1}{2} \left(x_n + \frac{a}{x_n} \right),$$

где a - фиксированное положительное число, а x_1 - любое положительное число.

Итерационная формула задаёт убывающую (начиная со 2-го элемента) последовательность, которая при любом выборе x_1 быстро сходится к искомой величине \sqrt{a} .

Данная программа вычисляет значение этой формулы до тех пор, пока её текущее значение не будет отличаться от её предыдущего значения менее чем на 0.05% (0,0005 в абсолютной величине).

В качестве случайного значения x_1 берётся число 3.

Входным параметром берётся целое положительное число.

2.2. Описание переменных программы и макросов

Строки:

- **strIntro** – строка приветствия
- **strInput** – пояснение для ввода числа **a**
- **strScan** – строка считывания числа
- **strError** – сообщение о неверном вводе
- **strValue** – вывод введённого пользователем числа **a**
- **strAnswer** – вывод ответа

Числа:

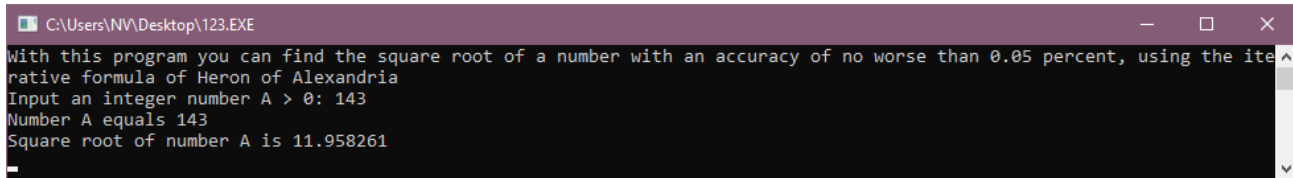
- **leftA** – левая граница ввода – число 0
- **x** – значение функции в момент итерации, изначально равно 3
- **a** – вводимое пользователем число
- **delta** – точность расчётов, равна 0.0005
- **prev** – значение функции в прошлой итерации
- **divideby2** – значение, равное 0.5

Макросы:

- **Print** – вывод строки в консоль
- **Scan** – считывание числа из консоли
- **GetAnswer** – Нахождение квадратного корня числа **a**

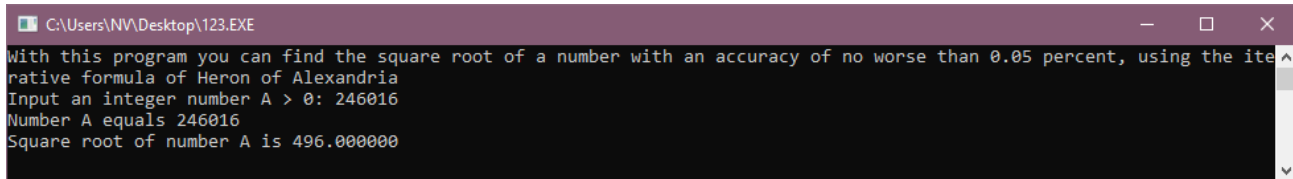
3. Тестирование программы

3.1. Корректные значения



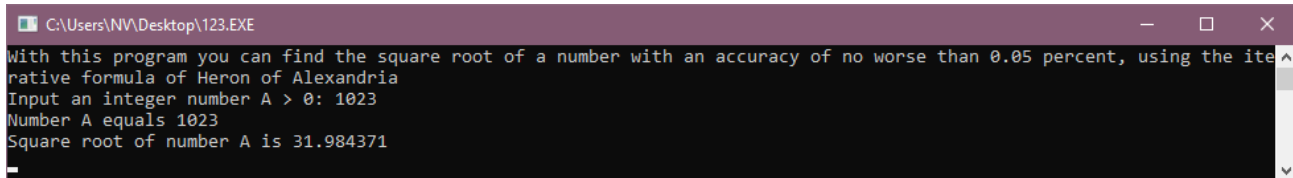
```
C:\Users\NV\Desktop\123.EXE
With this program you can find the square root of a number with an accuracy of no worse than 0.05 percent, using the iterative formula of Heron of Alexandria
Input an integer number A > 0: 143
Number A equals 143
Square root of number A is 11.958261
```

Рисунок 1. Число $a = 143$.



```
C:\Users\NV\Desktop\123.EXE
With this program you can find the square root of a number with an accuracy of no worse than 0.05 percent, using the iterative formula of Heron of Alexandria
Input an integer number A > 0: 246016
Number A equals 246016
Square root of number A is 496.000000
```

Рисунок 2. Число $a = 246016$.



```
C:\Users\NV\Desktop\123.EXE
With this program you can find the square root of a number with an accuracy of no worse than 0.05 percent, using the iterative formula of Heron of Alexandria
Input an integer number A > 0: 1023
Number A equals 1023
Square root of number A is 31.984371
```

Рисунок 3. Число $a = 1023$.

3.2. Некорректные значения

```

C:\Users\NV\Desktop\123.EXE
With this program you can find the square root of a number with an accuracy of no worse than 0.05 percent, using the iterative formula of Heron of Alexandria
Input an integer number A > 0: dfg
Number A must be integer, > 0 and < (2^31)-1!

```

Рисунок 4. Ввод букв.

```

C:\Users\NV\Desktop\123.EXE
With this program you can find the square root of a number with an accuracy of no worse than 0.05 percent, using the iterative formula of Heron of Alexandria
Input an integer number A > 0: -6
Number A must be integer, > 0 and < (2^31)-1!

```

Рисунок 5. Число a - отрицательное число.

```

C:\Users\NV\Desktop\123.EXE
With this program you can find the square root of a number with an accuracy of no worse than 0.05 percent, using the iterative formula of Heron of Alexandria
Input an integer number A > 0: 0
Number A must be integer, > 0 and < (2^31)-1!

```

Рисунок 6. Число a равно нулю.

```

C:\Users\NV\Desktop\123.EXE
With this program you can find the square root of a number with an accuracy of no worse than 0.05 percent, using the iterative formula of Heron of Alexandria
Input an integer number A > 0: 45.67
Number A equals 45
Square root of number A is 6.708204

```

Рисунок 7. Число a – вещественное число.

```

C:\Users\NV\Desktop\123.EXE
With this program you can find the square root of a number with an accuracy of no worse than 0.05 percent, using the iterative formula of Heron of Alexandria
Input an integer number A > 0: 4294967297
Number A equals 1
Square root of number A is 1.000000

```

Рисунок 8. Число a превышает 2^32.

ПРИЛОЖЕНИЕ 1

Список литературы

1. Список команд FPU. [Электронный ресурс] // URL: <https://prog-cpp.ru/asm-coprocessor-command/> (дата обращения: 31.10.2020)
2. Краткий перечень команд ассемблера. [Электронный ресурс] // URL: https://blic.fandom.com/ru/wiki/Краткий_перечень_команд_ассемблера (дата обращения: 31.10.2020)

Код программы

```
;Motorkin Vladimir BSE198
```

```
format PE console
```

```
entry start
```

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

```
section '.data' data readable writable
```

```
    strIntro db 'With this program you can find the square root of a number with  
an accuracy of no worse than 0.05 percent, using the iterative formula of Heron of  
Alexandria', 10, 0
```

```
    strInput db 'Input an integer number A > 0: ', 0
```

```
    strScan db '%d', 0
```

```
    strError db 'Number A must be integer, > 0 and < (2^31)-1!', 10, 0
```

```
    strValue db 'Number A equals %d', 10, 0
```

```
    strAnswer db 'Square root of number A is %1f', 10, 0
```

```
;strD db 'Current x = %1f'
```

```
leftA    dd 0
```

```
x        dq 3.0
```

```
a        dd ?
```

```
delta    dq 0.0005
```

```
prev     dq ?
```

```
divideby2 dd 0.5
```

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

```
macro Print [arg] {
```



```
reverse
    push arg
common
    call [printf]
}
```

```
macro Scan [args] {
    reverse
    push args
common
    call [scanf]
}
```

;All the calcilations for getting the answer

```
macro GetAnswer {
local valueLoop
```

```
valueLoop:
    ;debugging
    ;Print strD, dword[x], dword[x+4]

    ;previos value = x
    FLD [x]
    FSTP [prev]

    ;x = (x+a/x)*0.5
    FILD [a]
```

FDIV [x]

FADD [x]

FMUL [divideby2]

FSTP [x]

;checking if (x-prev)<delta, then repeat calculation

FLD [prev]

FSUB [x]

FABS

FCOMP [delta]

FSTSW ax

sahf

ja valueLoop

}

;-----Main-----

Error:

Print strError

jmp finish

start:

FINIT

Print strIntro

Print strInput

Scan strScan, a

xor ecx, ecx

mov ecx, [a]

cmp ecx, [leftA]

jle Error

Print strValue, [a]

GetAnswer

Print strAnswer, dword[x], dword[x+4]

finish:

call [getch]

push 0

call [ExitProcess]

;-----import-----

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'

include 'api\kernel32.inc'

import msvcrt,\

printf, 'printf',\

scanf, 'scanf',\

getch, '_getch'