

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ



МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
ВЫСШАЯ ШКОЛА ПЕЧАТИ И МЕДИАИНДУСТРИИ

*Факультет информационных технологий
Кафедра Информатики и информационных технологий*

направление подготовки

09.03.02 «Информационные системы и технологии»

ЛАБОРАТОРНАЯ РАБОТА № 3

Дисциплина: Интеллектуальные системы и технологии

Тема: Разработка модуля генерации и распознавания речи для естественно-языкового интерфейса интеллектуальной системы

Выполнил(а): студент(ка) группы 201-721

Лебедев А.М.

Дата, подпись 17.10.2022 _____
(Дата) (Подпись)

Проверил: _____
(Фамилия И.О., степень, звание) (Оценка)

Дата, подпись _____
(Дата) (Подпись)

Замечания: _____

Москва

Ход работы:**Задание 1. Квадратное уравнение**

Самостоятельно написать программу на Python вычисления корней квадратного уравнения $ax^2+bx+c=0$. Учесть все варианты значений коэффициентов a, b, c .

Листинг программы:

```
from cmath import inf
from math import sqrt
x1, x2, x, d = -inf, -inf, -inf, -inf
a, b, c = float(input()), float(input()), float(input())
d = b*b-4*a*c
if d==0:
    x = -b/(2*a)
    print("x = ", x)
elif d<0:
    print("Корней нет")
else:
    x1 = (-b + sqrt(d))/(2*a)
    x2 = (-b - sqrt(d))/(2*a)
    print('x1 = ', x1, "x2 = ", x2)
```

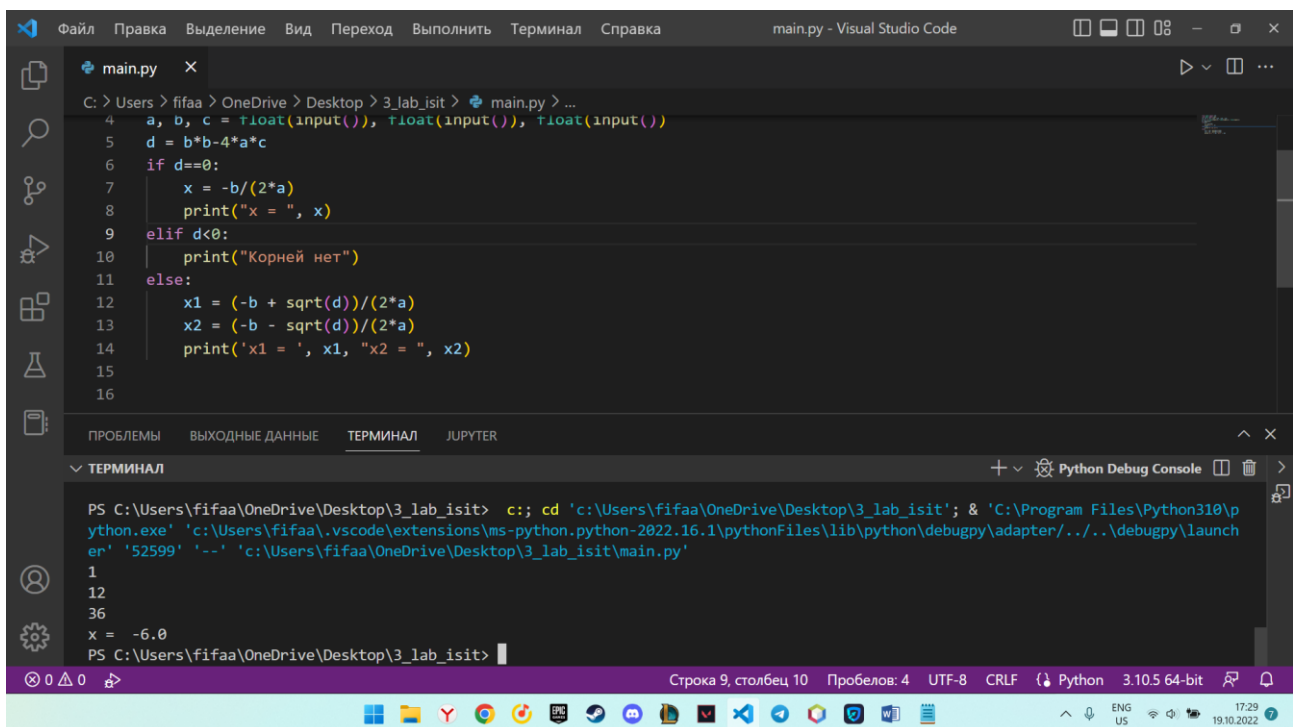


Рис. 1.1 – Пример работы программы с уравнением, имеющим 1 корень

```
main.py
C: > Users > fifaa > OneDrive > Desktop > 3_lab_isit > main.py > ...
1 from cmath import inf
2 from math import sqrt
3 x1, x2, x, d = -inf, -inf, -inf, -inf
4 a, b, c = float(input()), float(input()), float(input())
5 d = b*b-4*a*c
6 if d==0:
7     x = -b/(2*a)
8     print("x = ", x)
9 elif d<0:
10    print("Корней нет")
11 else:
12    x1 = (-b + sqrt(d))/(2*a)
13    x2 = (-b - sqrt(d))/(2*a)
```

```
PS C:\Users\fifaa\OneDrive\Desktop\3_lab_isit> c:; cd 'c:\Users\fifaa\OneDrive\Desktop\3_lab_isit'; & 'C:\Program Files\Python310\python.exe' 'c:\Users\fifaa\.vscode\extensions\ms-python.python-2022.16.1\pythonFiles\lib\python\debugpy\adapter\..\..\debugpy\launcher' '61800' '--' 'c:\Users\fifaa\OneDrive\Desktop\3_lab_isit\main.py'
1
4
-21
x1 = 3.0 x2 = -7.0
PS C:\Users\fifaa\OneDrive\Desktop\3_lab_isit>
```

Рис. 1.2 – Пример работы программы с уравнением, имеющим 2 корня

```
main.py
C: > Users > fifaa > OneDrive > Desktop > 3_lab_isit > main.py > ...
1 from cmath import inf
2 from math import sqrt
3 x1, x2, x, d = -inf, -inf, -inf, -inf
4 a, b, c = float(input()), float(input()), float(input())
5 d = b*b-4*a*c
6 if d==0:
7     x = -b/(2*a)
8     print("x = ", x)
9 elif d<0:
10    print("Корней нет")
11 else:
12    x1 = (-b + sqrt(d))/(2*a)
13    x2 = (-b - sqrt(d))/(2*a)
```

```
PS C:\Users\fifaa\OneDrive\Desktop\3_lab_isit> c:; cd 'c:\Users\fifaa\OneDrive\Desktop\3_lab_isit'; & 'C:\Program Files\Python310\python.exe' 'c:\Users\fifaa\.vscode\extensions\ms-python.python-2022.16.1\pythonFiles\lib\python\debugpy\adapter\..\..\debugpy\launcher' '56182' '--' 'c:\Users\fifaa\OneDrive\Desktop\3_lab_isit\main.py'
2
1
1
Корней нет
PS C:\Users\fifaa\OneDrive\Desktop\3_lab_isit>
```

Рис. 1.3 – Пример работы программы с уравнением, не имеющим корней

Задание 2. Тип треугольника

Самостоятельно написать программу на Python определения типа треугольника.

Учесть все возможные сочетания типов (например, равносторонний, равнобедренный-прямоугольный, равнобедренный-остроугольный и т.д.)

Листинг программы:

```
a, b, c = int(input()), int(input()), int(input())

if (a+b<=c or a+c<=b or b+c<=a):
    print("треугольника не существует")

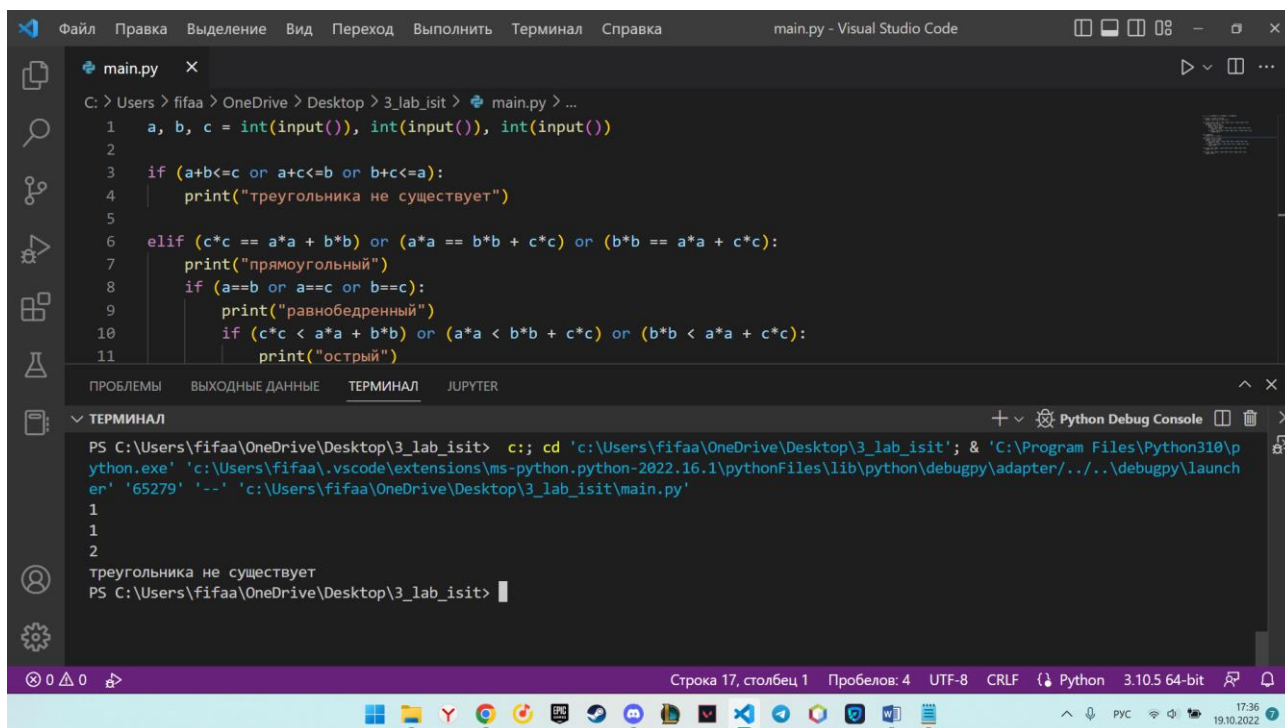
elif (c*c == a*a + b*b) or (a*a == b*b + c*c) or (b*b == a*a + c*c):
    print("прямоугольный")
    if (a==b or a==c or b==c):
        print("равнобедренный")

elif (a==b==c):
    print("равносторонний")

elif (a==b or a==c or b==c):
    print("равнобедренный")
    if (c*c < a*a + b*b) or (a*a < b*b + c*c) or (b*b < a*a + c*c):
        print("острый")
    elif (c*c > a*a + b*b) or (a*a > b*b + c*c) or (b*b > a*a + c*c):
        print("тупой")

elif (c*c < a*a + b*b) or (a*a < b*b + c*c) or (b*b < a*a + c*c):
    print("острый")

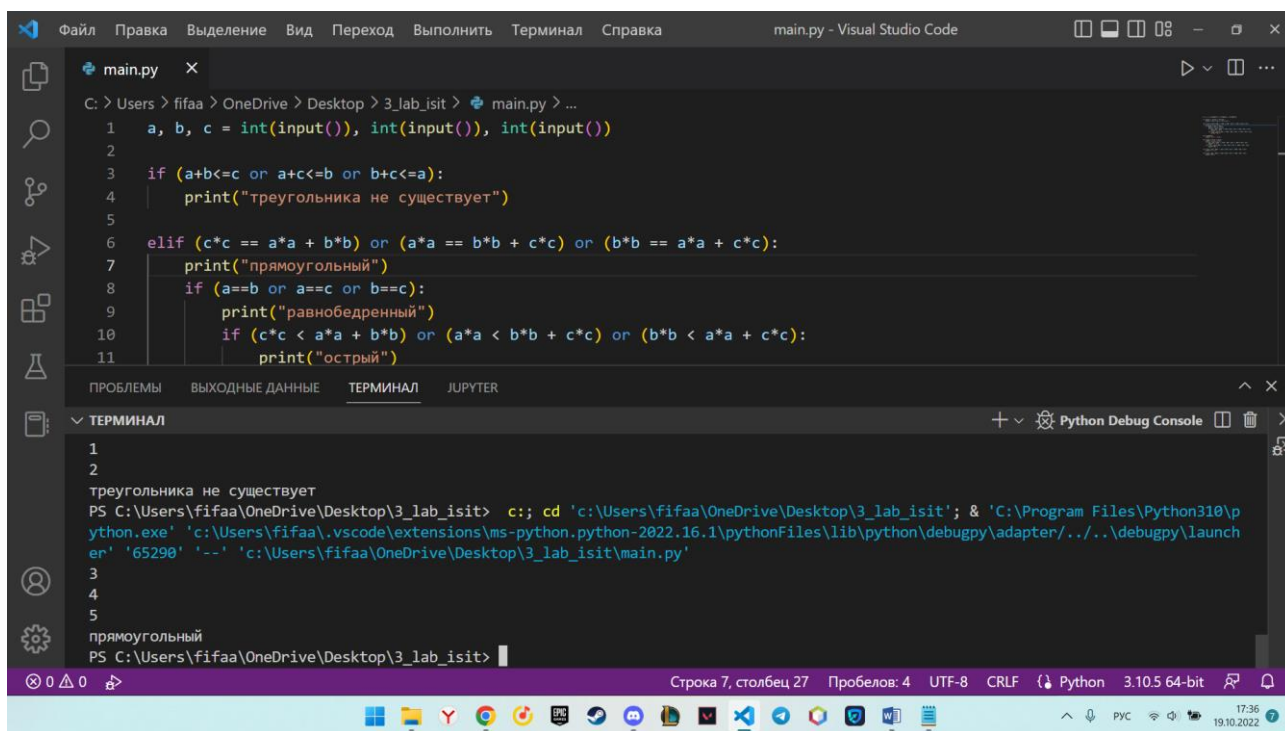
elif (c*c > a*a + b*b) or (a*a > b*b + c*c) or (b*b > a*a + c*c):
    print("тупой")
```



```
main.py X
C:\Users> fifaa > OneDrive > Desktop > 3_lab_isit > main.py > ...
1 a, b, c = int(input()), int(input()), int(input())
2
3 if (a+b<=c or a+c<=b or b+c<=a):
4     print("треугольника не существует")
5
6 elif (c*c == a*a + b*b) or (a*a == b*b + c*c) or (b*b == a*a + c*c):
7     print("прямоугольный")
8     if (a==b or a==c or b==c):
9         print("равнобедренный")
10        if (c*c < a*a + b*b) or (a*a < b*b + c*c) or (b*b < a*a + c*c):
11            print("острый")

ПРОБЛЕМЫ ВЫХОДНЫЕ ДАННЫЕ ТЕРМИНАЛ JUPYTER
▼ ТЕРМИНАЛ Python Debug Console
PS C:\Users\fifaa\OneDrive\Desktop\3_lab_isit> c:: cd 'c:\Users\fifaa\OneDrive\Desktop\3_lab_isit'; & 'C:\Program Files\Python310\python.exe' 'c:\Users\fifaa\.vscode\extensions\ms-python.python-2022.16.1\pythonFiles\lib\python\debugpy\adapter\..\debugpy\launcher' '65279' '--' 'c:\Users\fifaa\OneDrive\Desktop\3_lab_isit\main.py'
1
2
треугольника не существует
PS C:\Users\fifaa\OneDrive\Desktop\3_lab_isit>
```

Рис. 2.1 – Пример работы программы #1



```
main.py X
C:\Users> fifaa > OneDrive > Desktop > 3_lab_isit > main.py > ...
1 a, b, c = int(input()), int(input()), int(input())
2
3 if (a+b<=c or a+c<=b or b+c<=a):
4     print("треугольника не существует")
5
6 elif (c*c == a*a + b*b) or (a*a == b*b + c*c) or (b*b == a*a + c*c):
7     print("прямоугольный")
8     if (a==b or a==c or b==c):
9         print("равнобедренный")
10        if (c*c < a*a + b*b) or (a*a < b*b + c*c) or (b*b < a*a + c*c):
11            print("острый")

ПРОБЛЕМЫ ВЫХОДНЫЕ ДАННЫЕ ТЕРМИНАЛ JUPYTER
▼ ТЕРМИНАЛ Python Debug Console
1
2
треугольника не существует
PS C:\Users\fifaa\OneDrive\Desktop\3_lab_isit> c:: cd 'c:\Users\fifaa\OneDrive\Desktop\3_lab_isit'; & 'C:\Program Files\Python310\python.exe' 'c:\Users\fifaa\.vscode\extensions\ms-python.python-2022.16.1\pythonFiles\lib\python\debugpy\adapter\..\debugpy\launcher' '65298' '--' 'c:\Users\fifaa\OneDrive\Desktop\3_lab_isit\main.py'
3
4
5
прямоугольный
PS C:\Users\fifaa\OneDrive\Desktop\3_lab_isit>
```

Рис. 2.2 – Пример работы программы #2

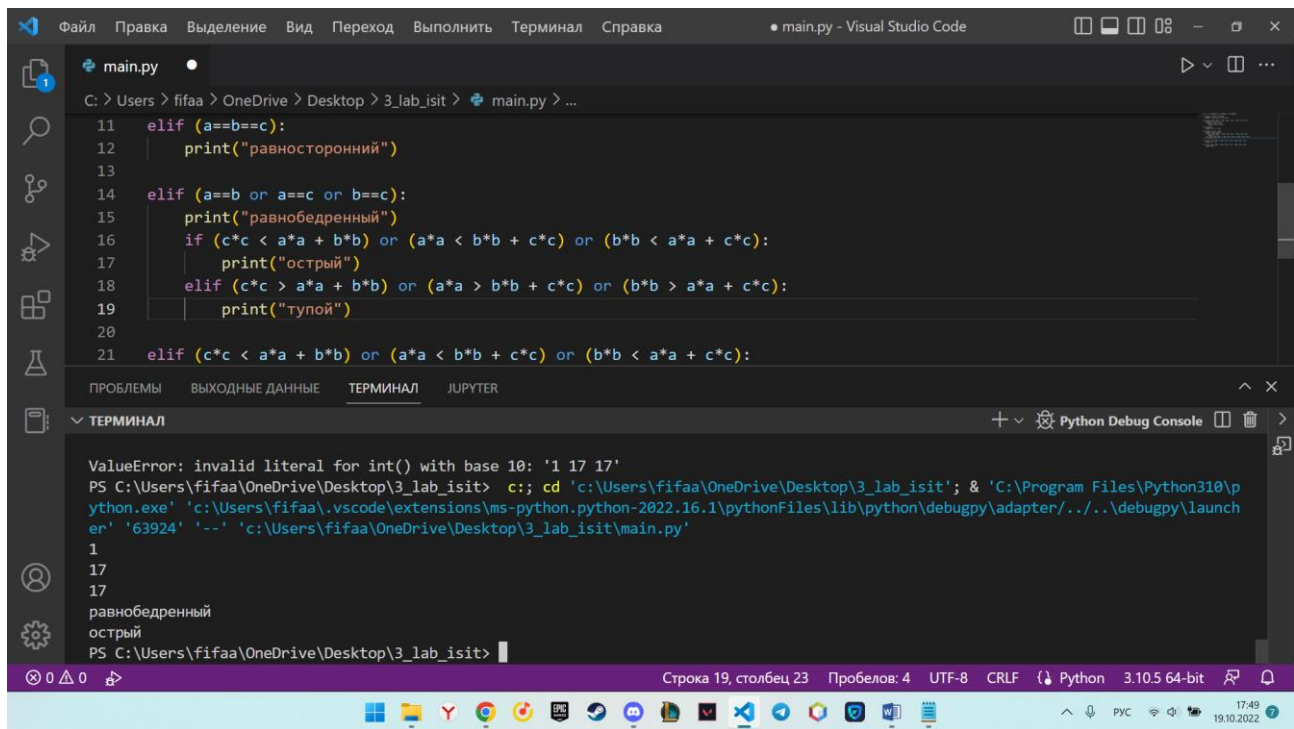


Рис. 2.3 – Пример работы программы #3

Задание 3. Табулирование функции

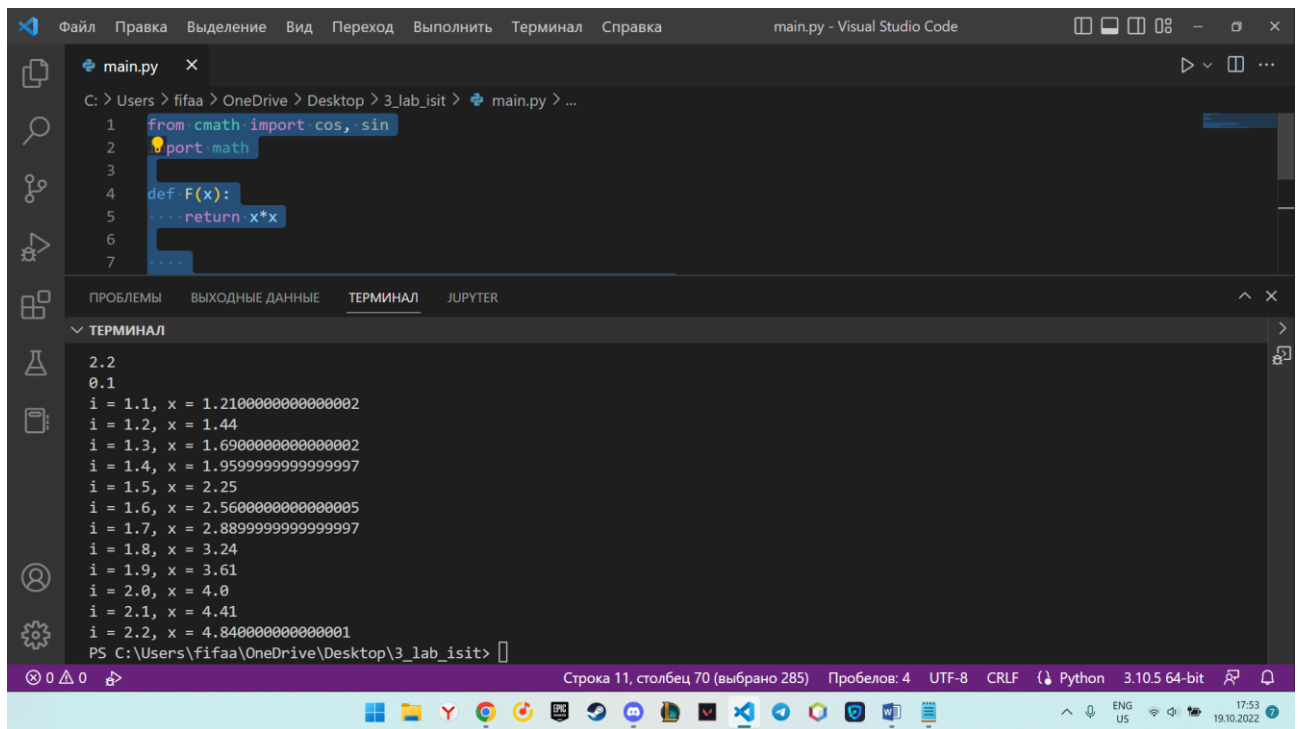
Самостоятельно написать программу на Python табулирования функции (выбрать функцию самостоятельно) на интервале $[A, B]$ с шагом H . Использовать форматированный вывод в Python.

Листинг программы:

```
from cmath import cos, sin
import math

def F(x):
    return x*x

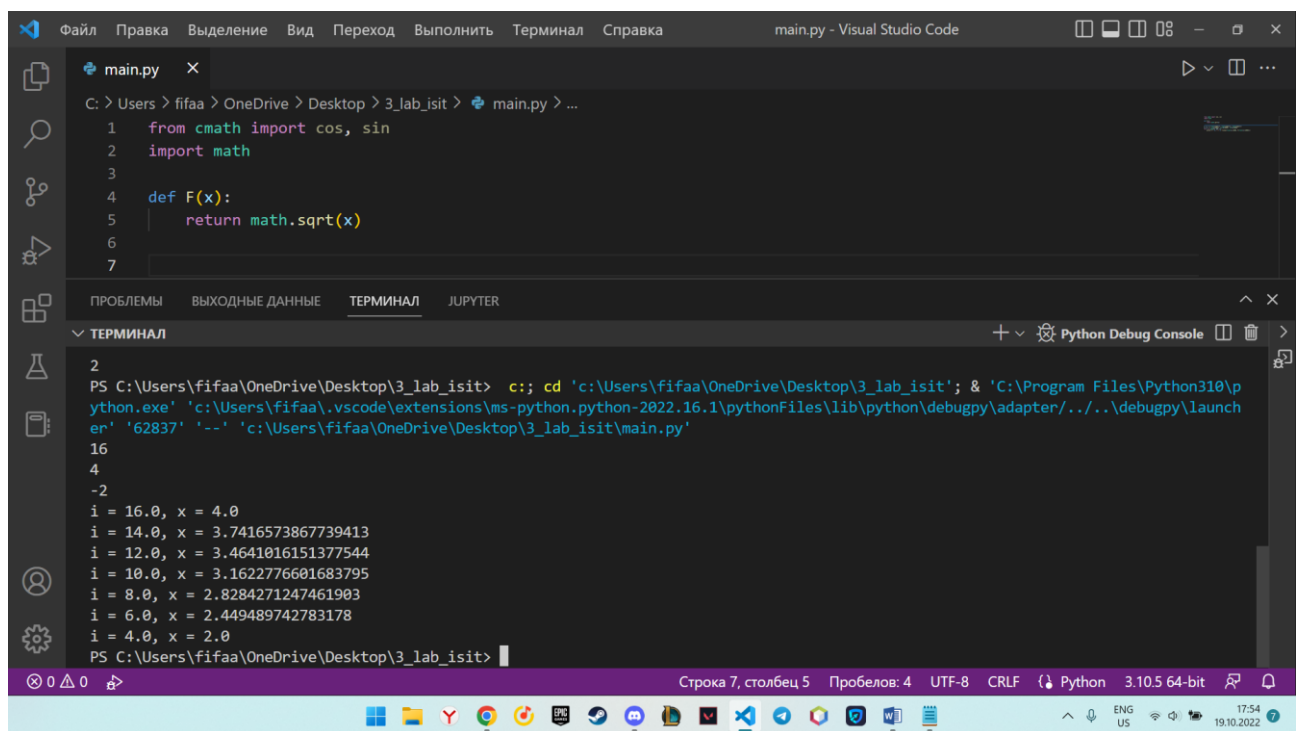
a, b, h = float(input()), float(input()), float(input())
a, b, h = int(a*10000), int(b*10000), int(h*10000)
for i in range(a, b+h, h):
    print("i = {}, x = {}".format(float(i)/10000, F(float(i)/10000)))
```

```
main.py
1 from cmath import cos, sin
2 import math
3
4 def F(x):
5     return x*x
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
```

```
2.2
0.1
i = 1.1, x = 1.2100000000000002
i = 1.2, x = 1.44
i = 1.3, x = 1.6900000000000002
i = 1.4, x = 1.9599999999999997
i = 1.5, x = 2.25
i = 1.6, x = 2.5600000000000005
i = 1.7, x = 2.8899999999999997
i = 1.8, x = 3.24
i = 1.9, x = 3.61
i = 2.0, x = 4.0
i = 2.1, x = 4.41
i = 2.2, x = 4.840000000000001
PS C:\Users\fifaa\OneDrive\Desktop\3_lab_isit>
```

Рис. 3.1 – Пример работы программы #1



```
main.py
1 from cmath import cos, sin
2 import math
3
4 def F(x):
5     return math.sqrt(x)
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
```

```
2
PS C:\Users\fifaa\OneDrive\Desktop\3_lab_isit> c:: cd 'c:\Users\fifaa\OneDrive\Desktop\3_lab_isit'; & 'C:\Program Files\Python310\python.exe' 'c:\Users\fifaa\.vscode\extensions\ms-python.python-2022.16.1\pythonFiles\lib\python\debugpy\adapter\..\..\debugpy\launcher' '62837' '--' 'c:\Users\fifaa\OneDrive\Desktop\3_lab_isit\main.py'
16
4
-2
i = 16.0, x = 4.0
i = 14.0, x = 3.7416573867739413
i = 12.0, x = 3.4641016151377544
i = 10.0, x = 3.1622776601683795
i = 8.0, x = 2.8284271247461903
i = 6.0, x = 2.449489742783178
i = 4.0, x = 2.0
PS C:\Users\fifaa\OneDrive\Desktop\3_lab_isit>
```

Рис. 3.2 – Пример работы программы #2

Задание 4. Модуль генерации и распознавания речи для естественно-языкового интерфейса интеллектуальной системы

Выполните 9 шагов, которые описываются во второй части "Руководства к практическому занятию № 3". Напишите и отладьте программу модуля.

Листинг программы:

```
import datetime
import sys
import webbrowser
import pyttsx3
import speech_recognition as sr

def talk(words):
    # print(words)
    # os.system("say "+words)
    engine = pyttsx3.init()
    engine.setProperty('rate', 150) # скорость речи
    engine.setProperty('volume', 1) # громкость (0-1)
    engine.say(words)
    engine.runAndWait()

def command():
    r = sr.Recognizer()
    with sr.Microphone() as source:
        print("Говорите !")
        talk("Говорите !")
        r.pause_threshold = 1 # пауза 1 сек
        r.adjust_for_ambient_noise(source, duration=1) # не слушаем шумы
        audio = r.listen(source)
    try:
        zadanie = r.recognize_google(audio, language="ru-RU").lower()
        print("Вы сказали: " + zadanie)
        talk("Вы сказали... " + zadanie)
        r.pause_threshold = 1 # пауза 1 сек
    except sr.UnknownValueError:
        talk("Не понимаю Вас!")
        zadanie = command()
    return zadanie

def time_to_text():#перевод времени в текст
    dict_hours = {1: 'час', 2: 'часа', 3: 'часа', 4: 'часа', 5: 'часов', 6:
'часов',
                  7: 'часов', 8: 'часов', 9: 'часов', 10: 'часов', 11: 'часов', 12:
'часов',
                  13: 'часов', 14: 'часов', 15: 'часов', 16: 'часов', 17: 'часов', 18:
'часов',
```



```

        19: 'часов', 20: 'часов', 21: 'час', 22: 'часа', 23: 'часа', 0:
'часов'}
    dict_minutes = {
        'минута': [1, 21, 31, 41, 51],
        'минуты': [2, 3, 4, 22, 23, 24, 32, 33, 34, 42, 43, 44, 52, 53, 54],
        'минут': [5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20,
                    25, 26, 27, 28, 29, 30,
                    35, 36, 37, 38, 39, 40,
                    45, 46, 47, 48, 49, 50,
                    55, 56, 57, 58, 59] }

    now = datetime.datetime.now()
    h = now.hour
    m = now.minute

    str_time = str(h) + dict_hours[h] + ' ... '

    for minutes in dict_minutes:
        if m in dict_minutes[minutes]:
            str_time += str(m) + ' ' + minutes
            break

    return str_time

def ParseZadanie(zadanie):
    """Разбор голосового задания/команды """
    if 'открой почту' in zadanie:
        talk('Хорошо, ОТКРЫВАЮ ПОЧТУ!')
        URL='https://mail.ru'
        webbrowser.open(URL)
    elif 'открой вк' in zadanie:
        talk('Хорошо, ОТКРЫВАЮ ВКОНТАКТЕ!')
        URL='https://vk.com/lebart1'
        webbrowser.open(URL)
    elif ('сколько времени' in zadanie) or ('который час' in zadanie) or ('сколько
время' in zadanie) :
        talk(time_to_text())
    elif ('как тебя зовут' in zadanie) or ('как твоё имя' in zadanie) or ('кто ты'
in zadanie):
        talk('Меня зовут Артемий! А как зовут тебя?')
    elif 'стоп' in zadanie:
        talk('Хорошо, заканчиваем разговор... До встречи !')
        sys.exit()

engine = pyttsx3.init()
voices = engine.getProperty('voices')
for voice in voices: # голоса и параметры каждого
    print('-----')
    print(f'Имя: {voice.name}')
    print(f'ID: {voice.id}')
    print(f'Язык(и): {voice.languages}')

```

```
print(f'Пол: {voice.gender}')
print(f'Возраст: {voice.age}')
for voice in voices:
    if 'Artemiy' in voice.name :
        engine.setProperty('voice', voice.id)

talk('Привет, меня зовут Артемий Лебедев! Давай поговорим!')

while True:
    ParseZadanie(command())
    talk("Поговорим еще ?")
```

https://drive.google.com/drive/folders/1f_W-tBAZNFGVSCRuwf-uEXJyhGB9DB-6?usp=sharing – ссылка на видео работы четвертого задания