

Лабораторная работа №1

Создание программы решения приведенного биквадратного уравнения.

```
import math

#Ввод данных с пользователем в консоль

d1 = True

d2 = True

d3 = True

print("Напишите первый коэффициент: ")

while d1:

    try:

        a = int(input())

        d1 = False

    except:

        print("Введенное значение некорректно. Попробуйте ввести новое значение:")

print("Напишите второй коэффициент: ")

while d2:

    try:

        b = int(input())

        d2 = False

    except:

        print("Введенное значение некорректно. Попробуйте ввести новое значение:")

print("Напишите третий коэффициент: ")

while d3:
```

```

try:
    c = int(input())
    d3 = False
except:
    print("Введенное значение некорректно. Попробуйте ввести новое значение:")
print("Вы ввели",a,b,c)

#Решение биквадратного уравнения через дискриминант, при замене  $t^2 = x$ 

#Проверка нулевых коэффициентов
if a == 0:
    if b == 0:
        print("Это не уравнение. Прошу,будьте серьезны")
    else:
        x = -c/b
        sqrX = math.sqrt(x)
        if x == 0:
            t1 = 0
            print("Корень один t =")
        else:
            t1 = sqrX
            t2 = -sqrX
            print("Два корня t1 = {} и t2 = {}".format(t1,t2))
elif b == 0:
    if c == 0:
        print("Корень t = 0")
    else:
        try:

```

```

    x = math.sqrt(-c/a)
    sqrX = math.sqrt(x)
    t1 = sqrX
    t2 = -sqrX
    print("Два корня t1 = {} и t2 = {}".format(t1,t2))
except:
    print("Нет корней")
elif c == 0:
    t1 = 0
    try:
        sqrX = math.sqrt(-a/b)
        t2 = sqrX
        t3 = -sqrX
        print("Три корня t1 = {}, t2 = {} и t3 = {}".format(t1,t2,t3))
    except:
        print("Корень лишь нулевой и он один")
#Решение через дискриминант при всех ненулевых коэффициентах
else:
    D = b**2 - 4*a*c
    print('Дискриминант равен:',D)
    if D > 0:
        x1 = (-b + math.sqrt(D))/(2*a)
        x2 = (-b - math.sqrt(D))/(2*a)
        if x1 > 0:
            t1 = math.sqrt(x1)
            t2 = -math.sqrt(x1)
            print('Первый и второй корень соответственно: t1=',t1,'t2=',t2)

```

```

else:
    print('Корень x1 не дает действительных корней')
if x2 > 0:
    t3 = math.sqrt(x2)
    t4 = -math.sqrt(x2)
    print('Еще два корня: t3=',t3,'t4=',t4)
else:
    print("Корень x2 не дает действительных корней")
elif D == 0:
    try:
        x = -b/(2*a)
    except:
        print('Беда')
if x == 0:
    print('Один корень: x1=', x)
elif x > 0:
    t1 = math.sqrt(x)
    t2 = -math.sqrt(x)
    print('Два конечных корня:',t1,t2)
else:
    print('Нет корней')

```

Лабораторная работа №2

```

from flask import Flask

```

```

from flask import request

```

```

app = Flask(__name__)

```

```

@app.route("/")

```

```
def hello_world():
    return "<p>Hello, World!</p>"

@app.route("/test")
def test():
    return "<p>Test</p>"

@app.route("/name/<user>")
def name(user):
    print(user)
    return "<p>Привет, {}</p>".format(user)

@app.route("/calc/sum/<a>/<b>")
def calc_sum(a, b):
    a=int(a)
    b=int(b)
    c = a + b
    return "<p>Сумма, {}</p>".format(c)

@app.route("/calc/sub")
def calc_sub():
    args_dict = request.args
    print(args_dict)
    a = float(args_dict["a"])
    b = float(args_dict["b"])
    c = a - b
    return "<p>Вычитание, {}</p>".format(c)

@app.route("/calc/multiple/<a>/<b>")
def calc_mult(a, b):
    a=int(a)
    b=int(b)
```

```
c = a * b

return "<p>Умножение, {}</p>".format(c)

@app.route("/calc/division/<a>/<b>")

def calc_div(a, b):

    a=int(a)

    b=int(b)

    c = a / b

    return "<p>Деление, {}</p>".format(c)

@app.route("/test/aboba")

def aboba_test():

    return "<p>Test na abobu uteryan</p>"

@app.route("/food")

def food():

    args_dict = request.args

    p = args_dict["первое"]

    v = args_dict["второе"]

    c = 0

    q = 0

    if p == "суп":

        c = 100

    elif p == "борщ":

        c = 150

    elif p == "щи":

        c = 50

    if v == "греча":

        q = 100

    elif v == "рыс":
```

```
q = 150
elif v == "пюрешка":
    q = 50
f = q + c
return "Стоимость за {} и {}: {}".format(p, v, f)
@app.route("/father/<name>")
def whoyourfather(name):
    return ''.format(name)
```

Лабораторная работа №3

Файл HTML:

```
<!doctype html>
<html>
  <head>
    <link rel = 'stylesheet' href="style.css">
    <script src="script.js"></script>
    <title>Лабораторная 3 Университет</title>
  </head>
  <body>
    <div class = "myheader">
      <h1>Кафедра СМ10</h1>
    </div>
    <div class = "myheader">
      <h2>Курс Веб-технологии</h2>
    </div>
    <div class = "myheader">
      <h3>Лабораторные</h3>
    </div>
```

```
<div class = "myheader">
```

```
  <ul>
```

```
    <li>Лаб 1</li>
```

```
    <li>Лаб 2</li>
```

```
    <li>Лаб 3</li>
```

```
  </ul>
```

```
</div>
```

```
<div id ='otv_face'>
```

```
  Ответственное лицо
```

```
  
```

```
</div>
```

```
<div>
```

```
  <ul>
```

```
    <li>Он убегает<br>
```

```
      </li>
```

```
    <li>Она догоняет<br>
```

```
      </li>
```

```
    <li>Он догоняет ту, что догоняет<br>
```

```
      <img src="./Tenma.webp"
```



```
        width = "400"
        alt="Дед не дома"></li>
    </ul>
</div>
<div>
    <table id="mytable">
        <tr>
            <th>Имя</th>
            <th>Вид</th>
            <th>Должность</th>
        </tr>
        <tr>
            <td>Димас</td>
            <td></li></td>
            <td>Он убегает</td>
        </tr>
        <tr>
            <td>Мадина</td>
            <td></li></td>
            <td>Она догоняет</td>
        </tr>
        <tr>
            <td>Макс</td>
```

```
<td></li></td>
<td>Он догоняет ту, что догоняет</td>
</tr>
</table>
</div>
```

Регистрация:

```
<br>
```

```
<label for = "name">
```

Имя:

```
</label>
```

```
<input id="in_name" type = "text" name = "name" value=""><br>
```

```
<label for = "name">
```

Возраст:

```
</label>
```

```
<input id = "in_age" type = "number" name = "age" value=""><br>
```

```
<label for = "name">
```

Пол:

```
</label>
```

```
<input id="in_sex" type = "radio" name = "sex" value="male" />
```

М

```
<input type = "radio" name = "sex" value="female" />
```

Ж

```
<br>
```

```
<label for = "name">
    Ответственный?
</label>

<input type = "checkbox" name = "otv">

<button type = "button" onclick =
"myfunction()">Зарегистрироваться</button>

<a href = "#otv_face">Ответственное лицо</a>

<a href = "https://animego.online/615-
monstr.html">Ugapanyuk</a>
```

```
</body>
</html>
```

Файл Css:

```
body{
    background-color: black;
    color: #AAbbcc;
}

#otv_face{
    background-color: red;
    color: white;
    width: 30%;
    margin: auto
}

.myheader{
    margin: 20px;
    font-family: "Courier New";
}
```

Файл Java:

```
function myfunction() {  
    n = document.getElementById("in_name").value;  
    a = document.getElementById("in_age").value;  
  
    response = "Имя: " + n + " Возраст: " + a  
    alert(response)  
    t = document.getElementById("mytable")  
    var row = t.insertRow(4);  
    var c_name = row.insertCell(0);  
    var c_photo = row.insertCell(1);  
    var c_task = row.insertCell(2);  
  
    c_name.innerHTML=n;  
    c_photo.innerHTML = ''  
    c_task.innerHTML="Боба"  
    alt="Дед не дома"  
}
```

Лабораторная работа №4

```
from flask import Flask, render_template, request  
from json import dumps as jsonstring
```

```
app = Flask(__name__)
```

```
class Classroom(object):  
    def __init__(self, name, teacher, id,num, image):  
        self.name = name
```

```
self.teach = teacher
```

```
self.floor = id
```

```
self.num = num
```

```
self.image = image
```

```
def __str__(self, name, teacher, floor,num, image):
```

```
    return("Название: ",name,
```

```
           " Преподаватель:", teacher,
```

```
           " Этаж:", floor,
```

```
           " Число студентов:", num,
```

```
           " Примерно выглядят: ", image)
```

```
class Kids(object):
```

```
    def __init__(self, name, classrooms, teacher):
```

```
        self.name = name
```

```
        self.classrooms = classrooms
```

```
        self.teacher = teacher
```

```
    def __str__(self, name, classrooms,teacher):
```

```
        return("Имя: ",name,
```

```
               " Класс:", classrooms,
```

```
               " Преподаватель:", teacher
```

```
        )
```

```
class_1b = Classroom("1A", "Самойлова Б.Б.", 1, 40, "puge.jpg")
```

```
class_2c = Classroom("2C", "Григорьев В.Д.", 2, 35, "shtib.jpg")
```

```
classes = [class_1b, class_2c]
```

```
student_a = Kids("ИВанова А.А.", classes, "Самойлова Б.Б.")
student_b = Kids("Бубова И.Г.", classes, "Григорьев В.Д.")
student_c = Kids("Ванов С.А.", classes, "Самойлова Б.Б.")
```

```
students = [student_a, student_b, student_c]
```

```
@app.route("/")
```

```
def hello_world():
```

```
    return render_template('index.html', student = student_a)
```

```
@app.route("/new_class")
```

```
def adding():
```

```
    name = request.args.get('name')
```

```
    floor = request.args.get('teacher')
```

```
    students = request.args.get('floor')
```

```
    zavkaf = request.args.get('num')
```

```
    new_class = Classroom(name, floor, students, zavkaf, "Anna liebert.jpeg")
```

```
    student_a.classrooms.append(new_class)
```

```
    return "Добавил"
```