



**TATENDA  
GANYANI**

Junior mathematics teacher

## CONTACT



Ulitsa Mkhluho Maklaya dom 17 Russia



tgbmhumhi5@gmail.com



+7 962 911 27 36

## INTERESTS

- ✓ **Sports**, football
- ✓ **Passions**, solving real world mathematical problems
- ✓ **Projects**, polynomial interpolation, solving systems of linear equations ,
- ✓ Solving initial value problems, cubic splines with python

## LANGUAGES

English

Russian

## SKILLS

python

javascript

Node js

react js



## EDUCATION

2018- 2023

Moscow Russia

**Level of study: Undergraduate third year**

*University, People's Friendship University*

*Program : Applied Mathematics and Informatics*



## WORK EXPERIENCE

- Linear Algebra
- Mathematical Analysis
- Theory of differential equations
- Complex Analysis
- Functional Analysis
- Numerical Analysis
- Equations of mathematical physics

### One of my previous projects

Newton's polynomial interpolation with python

```
In [ ]: #Интерполяционный многочлен Ньютона
```

```
[319]: import numpy as np
import matplotlib.pyplot as plt
import sympy as sym
```

```
[320]: x = sym.Symbol('x')
```

```
[321]: X = np.array([-np.pi/2, -np.pi/4, 0, np.pi/4, np.pi/2])
Y = np.array([-0.5, -(1/2)**(1/2), 0, (1/2)**(1/2), 0.5])
```

```
[322]: def polinom_newtona(X,Y):
    suma = 0
    for i in range(0, len(X)):
        f_x=0
        L2_x=1
        for j in range(0,i+1):
            L1_x=1
            for k in range(0,i+1):
                if k!=j:
                    L1_x*=X[j]-X[k]
            f_x+=(Y[j]/L1_x)
            if j!=i:
                L2_x*=x-X[j]
            suma+=(f_x*L2_x)
    return suma
```

```
[323]: S= polinom_newtona(X,Y)
```

```
In [333]: plt.plot(X1,Y1,X1,Y2)
plt.legend(['многочлен Ньютона', 'cos(x)'])
```

```
Out[333]: <matplotlib.legend.Legend at 0x2742c23c4a8>
```

