

МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

им. Н.Э. Баумана

Факультет “Информатика и системы управления” Кафедра “Системы
обработки информации и управления”



Дисциплина “Парадигмы и конструкции языков программирования” Отчет по
Лабораторное работе №1

Выполнил:

Студент группы ИУ5-35Б

Костылев М.С.

Преподаватель:

Гапанюк Ю.Е.

Москва 2025

Задание:

Разработать программу для решения [биквадратного уравнения](#).

1. Программа должна быть разработана в виде консольного приложения на языке Python.
2. Программа осуществляет ввод с клавиатуры коэффициентов A, B, C, вычисляет дискриминант и **ДЕЙСТВИТЕЛЬНЫЕ** корни уравнения (в зависимости от дискриминанта).
3. Коэффициенты A, B, C могут быть заданы в виде параметров командной строки ([вариант задания параметров приведен в конце файла с примером кода](#)). Если они не заданы, то вводятся с клавиатуры в соответствии с пунктом 2. [Описание работы с параметрами командной строки](#).
4. Если коэффициент A, B, C введен или задан в командной строке некорректно, то необходимо проигнорировать некорректное значение и вводить коэффициент повторно пока коэффициент не будет введен корректно. Корректно заданный коэффициент - это коэффициент, значение которого может быть без ошибок преобразовано в действительное число.
5. Дополнительное задание 1 (*). Разработайте две программы на языке Python - одну с применением процедурной парадигмы, а другую с применением объектно-ориентированной парадигмы.
6. Дополнительное задание 2 (*). Разработайте две программы - одну на языке Python, а другую на любом другом языке программирования (кроме C++).

Код программы:

```
"""
```

Программа для решения биквадратного уравнения вида: $Ax^4 + Bx^2 + C = 0$

```
"""
```

```
import sys
```

```
import math
```

```
def get_valid_coefficient(prompt, value=None):
```

```
    if value is not None:
```

```
        try:
```

```
            coefficient = float(value)
```

```
            print(f'Коэффициент из командной строки: {coefficient}')
```

```
            return coefficient
```

```
        except ValueError:
```

```
print(f'Некорректное значение коэффициента из командной строки:
'{value}'. Требуется ввод с клавиатуры.")
```

```
while True:
```

```
    try:
```

```
        user_input = input(prompt)
```

```
        coefficient = float(user_input)
```

```
        return coefficient
```

```
    except ValueError:
```

```
        print("Ошибка! Введите действительное число.")
```

```
def solve_biquadratic(a, b, c):
```

```
    if a == 0:
```

```
        if b == 0:
```

```
            if c == 0:
```

```
                print("Уравнение  $0 = 0$  имеет бесконечно много решений")
```

```
                return ["Любое действительное число"]
```

```
            else:
```

```
                print(f"Уравнение  $\{c\} = 0$  не имеет решений")
```

```
                return []
```

```
        else:
```

```
            print(f"Уравнение становится квадратным:  $\{b\}x^2 + (\{c\}) = 0$ ")
```

```
            return solve_quadratic_for_t(b, c, a)
```

```
print(f'\nРешаем биквадратное уравнение: ', end="")
```

```
terms = []
```

```
if a != 0:
```

```

terms.append(f"{a}x^4")
if b != 0:
    sign = "+" if b > 0 else ""
    terms.append(f"{sign}{b}x^2")
if c != 0:
    sign = "+" if c > 0 else ""
    terms.append(f"{sign}{c}")
print(" ".join(terms) + " = 0")

```

```
D = b**2 - 4*a*c
```

```
print(f"Дискриминант  $D = B^2 - 4AC = \{b\}^2 - 4*\{a\}*\{c\} = \{D\}$ ")
```

```

if D < 0:
    print("D < 0: Уравнение не имеет действительных корней")
    return []

```

```

sqrt_D = math.sqrt(D)
t1 = (-b + sqrt_D) / (2*a)
t2 = (-b - sqrt_D) / (2*a)

```

```
print(f"Корни для  $t = x^2$ :")
```

```
print(f"  $t_1 = (-B + \sqrt{D}) / (2A) = (\{-b\} + \{sqrt\_D:.3f\}) / (\{2*a\}) = \{t1:.3f\}$ ")
```

```
print(f"  $t_2 = (-B - \sqrt{D}) / (2A) = (\{-b\} - \{sqrt\_D:.3f\}) / (\{2*a\}) = \{t2:.3f\}$ ")
```

```
real_roots = []
```

if $t_1 > 0$:

$x_1 = \text{math.sqrt}(t_1)$

$x_2 = -\text{math.sqrt}(t_1)$

`real_roots.extend([x1, x2])`

`print(f't1 = {t1:.3f} > 0, дает два корня: $x_1 = \{x_1:.3f\}$, $x_2 = \{x_2:.3f\}$ ')`

elif $t_1 == 0$:

$x = 0.0$

`real_roots.append(x)`

`print(f't1 = {t1:.3f} = 0, дает один корень: $x = \{x:.3f\}$ ')`

else:

`print(f't1 = {t1:.3f} < 0, не дает действительных корней')`

if $t_2 \neq t_1$:

if $t_2 > 0$:

$x_3 = \text{math.sqrt}(t_2)$

$x_4 = -\text{math.sqrt}(t_2)$

if not any(`math.isclose(x3, root, rel_tol=1e-9)` for root in `real_roots`):

`real_roots.extend([x3, x4])`

`print(f't2 = {t2:.3f} > 0, дает два корня: $x_3 = \{x_3:.3f\}$, $x_4 = \{x_4:.3f\}$ ')`

elif $t_2 == 0$ and 0 not in `real_roots`:

$x = 0.0$

`real_roots.append(x)`

`print(f't2 = {t2:.3f} = 0, дает один корень: $x = \{x:.3f\}$ ')`

elif $t_2 < 0$:

`print(f't2 = {t2:.3f} < 0, не дает действительных корней')`

```
return sorted(real_roots)
```

```
def solve_quadratic_for_t(b, c, a):
```

```
    print(f'\nРешаем квадратное уравнение:  $\{b\}x^2 + (\{c\}) = 0$ ')

```

```
    if b == 0:
```

```
        if c == 0:
```

```
            print("Уравнение  $0 = 0$  имеет бесконечно много решений")
```

```
            return ["Любое действительное число"]
```

```
        else:
```

```
            print(f'Уравнение  $\{c\} = 0$  не имеет решений')
```

```
            return []

```

```
    if -c/b < 0:
```

```
        print(f' $x^2 = \{-c\}/\{b\} = \{-c/b:.3f\} < 0$ , уравнение не имеет действительных корней')
```

```
        return []

```

```
    x_squared = -c/b
```

```
    if x_squared == 0:
```

```
        x = 0.0
```

```
        print(f' $x^2 = \{x\_squared\}$ , уравнение имеет один корень:  $x = \{x:.3f\}$ ')

```

```
        return [x]

```

```
    x1 = math.sqrt(x_squared)
```

```

x2 = -x1

print(f'x2 = {-c}/{b} = {x_squared:.3f}', уравнение имеет два корня: x1 =
{x1:.3f}, x2 = {x2:.3f}')

return sorted([x1, x2])

def main():
    """Основная функция программы"""
    print("=" * 60)
    print("РЕШЕНИЕ БИКВАДРАТНОГО УРАВНЕНИЯ: Ax4 + Bx2 + C = 0")
    print("=" * 60)

    coefficients_from_cli = [None, None, None]

    if len(sys.argv) > 1:
        coefficients_from_cli[0] = sys.argv[1] if len(sys.argv) > 1 else None
    if len(sys.argv) > 2:
        coefficients_from_cli[1] = sys.argv[2] if len(sys.argv) > 2 else None
    if len(sys.argv) > 3:
        coefficients_from_cli[2] = sys.argv[3] if len(sys.argv) > 3 else None

    print("\nВведите коэффициенты биквадратного уравнения Ax4 + Bx2 + C =
0")

    print("(Коэффициенты можно задать как параметры командной строки)")

    a = get_valid_coefficient("Введите коэффициент A: ",
coefficients_from_cli[0])

```

```
b = get_valid_coefficient("Введите коэффициент В: ",
coefficients_from_cli[1])
```

```
c = get_valid_coefficient("Введите коэффициент С: ",
coefficients_from_cli[2])
```

```
roots = solve_biquadratic(a, b, c)
```

```
print("\n" + "=" * 60)
```

```
print("РЕЗУЛЬТАТЫ:")
```

```
print("=" * 60)
```

```
if not roots:
```

```
    print("Уравнение не имеет действительных корней")
```

```
elif roots == ["Любое действительное число"]:
```

```
    print("Уравнение имеет бесконечно много решений:")
```

```
    print("  $x \in \mathbb{R}$  (любое действительное число)")
```

```
else:
```

```
    if len(roots) == 1:
```

```
        print(f"Уравнение имеет 1 действительный корень:")
```

```
        print(f"  $x = \{roots[0]:.6f\}$ ")
```

```
    else:
```

```
        print(f"Уравнение имеет {len(roots)} действительных корня:")
```

```
        for i, root in enumerate(roots, 1):
```

```
            print(f"  $x_{\{i\}} = \{root:.6f\}$ ")
```

```
print("\nПроверка корней:")
```



```
for root in roots:
```

```
    if root != "Любое действительное число":
```

```
        result = a * root**4 + b * root**2 + c
```

```
        print(f" Для x = {root:.6f}:  $A \cdot x^4 + B \cdot x^2 + C = {result:.10f} \approx 0$ ")
```

```
if __name__ == "__main__":
```

```
    main()
```

Работа программы:

```

spiral@LAPTOP-L1VU5G9D:~/Kostylev_3SEM_25/RK2$ cd ..
spiral@LAPTOP-L1VU5G9D:~/Kostylev_3SEM_25$ cd labs
spiral@LAPTOP-L1VU5G9D:~/Kostylev_3SEM_25/labs$ python3 ./biquadratic_equation.py
=====
РЕШЕНИЕ БИКВАДРАТНОГО УРАВНЕНИЯ:  $Ax^4 + Bx^2 + C = 0$ 
=====

Введите коэффициенты биквадратного уравнения  $Ax^4 + Bx^2 + C = 0$ 
(Коэффициенты можно задать как параметры командной строки)
Введите коэффициент A: 9
Введите коэффициент B: 8
Введите коэффициент C: -1

Решаем биквадратное уравнение:  $9.0x^4 + 8.0x^2 - 1.0 = 0$ 
Дискриминант  $D = B^2 - 4AC = 8.0^2 - 4*9.0*-1.0 = 100.0$ 
Корни для  $t = x^2$ :
 $t_1 = (-B + \sqrt{D}) / (2A) = (-8.0 + 10.000) / (18.0) = 0.111$ 
 $t_2 = (-B - \sqrt{D}) / (2A) = (-8.0 - 10.000) / (18.0) = -1.000$ 
 $t_1 = 0.111 > 0$ , дает два корня:  $x_1 = 0.333$ ,  $x_2 = -0.333$ 
 $t_2 = -1.000 < 0$ , не дает действительных корней

=====
РЕЗУЛЬТАТЫ:
=====
Уравнение имеет 2 действительных корня:
 $x_1 = -0.333333$ 
 $x_2 = 0.333333$ 

Проверка корней:
Для  $x = -0.333333$ :  $A \cdot x^4 + B \cdot x^2 + C = -0.0000000000 \approx 0$ 
Для  $x = 0.333333$ :  $A \cdot x^4 + B \cdot x^2 + C = -0.0000000000 \approx 0$ 

```

```

spiral@LAPTOP-L1VU5G9D:~/Kostylev_3SEM_25/labs$ python3 ./biquadratic_equation.py
=====
РЕШЕНИЕ БИКВАДРАТНОГО УРАВНЕНИЯ:  $Ax^4 + Bx^2 + C = 0$ 
=====

Введите коэффициенты биквадратного уравнения  $Ax^4 + Bx^2 + C = 0$ 
(Коэффициенты можно задать как параметры командной строки)
Введите коэффициент A: 0
Введите коэффициент B: 4
Введите коэффициент C: 4
Уравнение становится квадратным:  $4.0x^2 + (4.0) = 0$ 

Решаем квадратное уравнение:  $4.0x^2 + (4.0) = 0$ 
 $x^2 = -4.0/4.0 = -1.000 < 0$ , уравнение не имеет действительных корней

=====
РЕЗУЛЬТАТЫ:
=====
Уравнение не имеет действительных корней

Проверка корней:

```