

Міністерство освіти і науки України
Національний технічний університет України
«Київський політехнічний інститут»
Фізико-технічний інститут

ЛАБОРАТОРНА РОБОТА №1

З предмету
«Алгоритми та структури даних»

Виконав:
Студент ФІ-23 групи
Булавінцев Юрій

Київ 2024

Варіант № 1

Мета роботи: отримати навички роботи з рекурсивними алгоритмами.

Завдання:

Виконати наступні завдання:

1. Проаналізувати умову задачі.
2. Розробити алгоритм та створити програму розв'язання задачі згідно з номером варіанту.
3. Результати роботи оформити протоколом.

Завдання для лабораторної роботи №1

Створити збережену процедуру, вказану в завданні.

1. Піднести до додатного цілого степеня дійсне ненульове число.
2. Знайти НСД двох цілих чисел за алгоритмом Евкліда.
3. Реалізувати алгоритм для розв'язання задачі «Ханойські вежі». Виписати послідовність ходів для перекладання n дисків вежі ($n = 2; 3; 4; 5$ дисків, використати онлайн гру).

1. Піднести до додатного цілого степеня дійсне ненульове число

```
def step(b, a):  
    if a == 0:  
        return 1  
    else:  
        return b*step(b, a-1)  
  
a = float(input())  
b = float(input())  
i = a  
if a > -1 and a % 1 == 0 and b != 0:
```

```

print(step(b, a))
elif a < 0 or a % 1 != 0 and b == 0:
    print("Введено не ціле додатнє число і число b дорівнює 0" )
elif b == 0:
    print("Число b дорівнює 0")
elif a < 0 or a % 1 != 0:
    print("Введено не ціле додатнє число a")

```

2. Знайти НСД двох цілих чисел за алгоритмом Евкліда

```

def nsd(a,b):
    if a!=0 and b!=0:
        if a>b:
            a=a%b
            return nsd(a,b)
        else:
            b=b%a
            return nsd(a,b)
    else:
        return a+b
a=int(input())
b=int(input())
a=abs(a)
b=abs(b)
print(nsd(a,b))

```

3. Реалізувати алгоритм для розв'язання задачі «Ханойські вежі». Виписати послідовність ходів для перекладання n дисків вежі (n = 2; 3; 4; 5 дисків, використати онлайн гру).

```

def hanoy(a, b, c, d):
    if d == 1:
        print(a, "->", c)
    else:
        hanoy(a, c, b, d - 1)
        print(a, "->", c)
        hanoy(b, a, c, d - 1)

```

```

hanoy("1", "2", "3", 4)

```

Для n=2: 1 -> 2, 1 -> 3 2 -> 3.

Для n=3: 1 -> 3, 1 -> 2, 3 -> 2, 1 -> 3, 2 -> 1, 2 -> 3, 1 -> 3.

Для n=4: 1 -> 2, 1 -> 3, 2 -> 3, 1 -> 2, 3 -> 1, 3 -> 2, 1 -> 2, 1 -> 3, 2 -> 3, 2 -> 1, 3 -> 1, 2 -> 3, 1 -> 2, 1 -> 3, 2 -> 3.

Для n=5: 1 -> 3, 1 -> 2, 3 -> 2, 1 -> 3, 2 -> 1, 2 -> 3, 1 -> 3, 1 -> 2, 3 -> 2, 3 -> 1, 2 -> 1, 3 -> 2, 1 -> 3, 1 -> 2, 3 -> 2, 1 -> 3, 2 -> 1, 2 -> 3, 2 -> 1, 2 -> 3, 1 -> 3, 2 -> 1, 3 -> 2, 3 -> 1, 2 -> 1, 2 -> 3, 1 -> 3, 1 -> 2, 3 -> 2, 1 -> 3, 2 -> 1, 2 -> 3, 1 -> 3.