

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Лабораторна робота №5
Дисципліна: Комп'ютерна дискретна математика

Виконав
Студент групи ПЗПІ-23-3
Харченко Федір Олександрович

Перевірив
Старший викладач каф. ПІ
Терещенко Гліб Юрійович

Тема: "Function Analysis"

Мета: Understand and implement the Function Analysis operations operations

Код програми:

<https://github.com/rintaro129/CDM/tree/main/LAB5>

```

1  import sympy
2  class Lab5:
3      def functionEvaluator(m : int, b : int, x : int) -> int:
4          return m * x + b
5
6
7      def domainAndRange(s : set):
8          dom = set()
9          ran = set()
10         for (a, b) in s:
11             dom.add(a)
12             ran.add(b)
13         return dom, ran
14
15     def evenOdd(s : list) -> str:
16         s = set(s)
17         ifEven = True
18         ifOdd = True
19         for (a, b) in s:
20             if(ifEven):
21                 if((-a, b) not in s):
22                     ifEven = False
23             if(ifOdd):
24                 if((-a, -b) not in s):
25                     ifOdd = False
26             if(not ifOdd and not ifEven):
27                 break
28         if(ifEven):
29             return "Even"
30         if(ifOdd):
31             return "Odd"
32         return "Neither"
33
34
35     def injectiveFunctionValidator(s : set) -> bool:
36         res = True
37         ran = set()
38         for (_, b) in s:
39             if(b in ran):
40                 res = False
41                 break
42             else:
43                 ran.add(b)
44         return res
45
46     def surjectiveFunctionValidator(s : set, c : set) -> bool:
47         ran = set()
48         for (_, b) in s:
49             ran.add(b)
50         if(ran == c):
51             return True
52         else:
53             return False
54

```

```

55     def funcCombTool(func1 : str, func2 : str, operation : str, value) -> int:
56         x = sympy.Symbol('x')
57         func1 = eval(func1.replace('^', ' ** '))
58         func2 = eval(func2.replace('^', ' ** '))
59         res1 = func1.subs(x, value)
60         res2 = func2.subs(x, value)
61         if(operation == "Addition"):
62             return res1 + res2
63         elif(operation == "Subtraction"):
64             return res1 - res2
65         elif(operation == "Multiplication"):
66             return res1 * res2
67         elif(operation == "Division"):
68             return res1 / res2
69         else:
70             print("Unsupported operation")
71             return 0
72
73
74     def graphInformationExtractor(s : list):
75         s = sorted(s)
76         xintercepts = set()
77         yintercepts = set()
78         maxima = set()
79         minima = set()
80         for (a, b) in s:
81             if(a == 0):
82                 xintercepts.add(b)
83             if(b == 0):
84                 yintercepts.add(a)
85         d = 0
86         if(s[1][1] - s[0][1] != 0):
87             d = (s[1][1] - s[0][1]) / abs(s[1][1] - s[0][1])
88         for i in range(2, len(s)):
89             d1 = 0
90             if(s[i][1] - s[i-1][1] != 0):
91                 d1 = (s[i][1] - s[i-1][1]) / abs(s[i][1] - s[i-1][1])
92             if(d == 0):
93                 d = d1
94                 continue
95             if(d == 1 and d1 == -1):
96                 d = -1
97                 maxima.add(s[i-1])
98             if(d == -1 and d1 == 1):
99                 d = 1
100                 minima.add(s[i-1])
101         print(f"x-intercepts: {xintercepts}")
102         print(f"y-intercepts: {yintercepts}")
103         print(f"Maxima: {maxima}")
104         print(f"Minima: {minima}")
105         return xintercepts, yintercepts, maxima, minima
106
107     print(Lab5.functionEvaluator(m=2, b=3, x=4))
108     dom, ran = Lab5.domainAndRange([(1, 2), (3, 6), (4, 8)])
109     print(f"Domain: {dom}\nRange: {ran}")
110     print(Lab5.evenOdd([(-1,1), (0,0), (1,1)]))
111     print(Lab5.injectiveFunctionValidator([(2, 4), (3, 6), (4, 8)]))
112     print(Lab5.surjectiveFunctionValidator([(1, 2), (2, 3), (3, 4)], {2, 3, 4}))
113     print(Lab5.funcCombTool("x ^ 2", "2*x+1", "Addition", 3))
114     Lab5.graphInformationExtractor([(-2, -4), (-1, -1), (0, 0), (1, 1), (2, 4)])
115

```

```
11
Domain: {1, 3, 4}
Range: {8, 2, 6}
Even
True
True
16
x-intercepts: {0}
y-intercepts: {0}
Maxima: set()
Minima: set()
```

Висновок: All tasks are implemented successfully.