Федеральное государственное образовательное бюджетное

учреждение высшего образования

«**Финансовый университет**

**при Правительстве Российской Федерации»**

**(Финансовый университет)**

**Департамент анализа данных, принятия решений и финансовых технологий**

**Отчёт по семинару №4**

**Выполнил:**

Студент Малкеров Г.А.

Группы ЗБ-ПИ19-2

Вариант 10

**Проверил:**

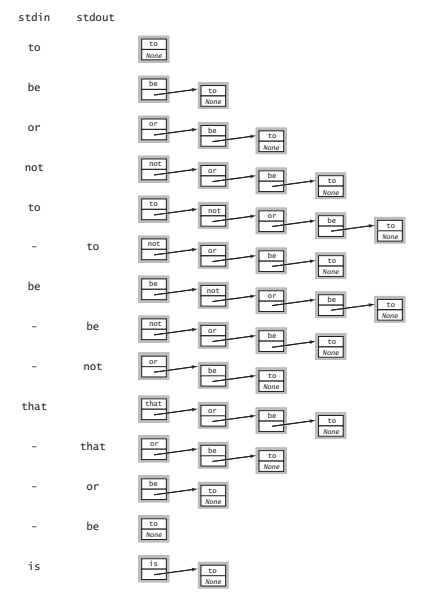
Бочаров М.И.

**Москва 2020**

**Задание:**

****

**Реализация схемы трасировки работы программы:**

****

**Исходный код наисанной программы dedup.py:**

import stdio

import sys

class Stack:

def \_\_init\_\_(self):

self.\_first = None

def isEmpty(self):

return self.\_first is None

def push(self, item):

self.\_first = \_Node(item, self.\_first)

def pop(self):

item = self.\_first.item

self.\_first = self.\_first.next

return item

def \_\_str\_\_(self):

s = ''

cur = self.\_first

while cur is not None:

s += str(cur.item) + ' '

cur = cur.next

return s

class \_Node:

def \_\_init\_\_(self, item, next):

self.item = item

self.next = next

def tracer(frame, event, arg):

print(event, frame.f\_lineno, frame.f\_locals)

return tracer

def tracerNull(frame, event, arg):

return tracerNull

def main():

stack = Stack()

while not stdio.isEmpty():

item = stdio.readString()

sys.settrace(tracer)

if item != '-':

stack.push(item)

else:

stdio.write(stack.pop() + ' ')

sys.settrace(tracerNull)

stdio.writeln()

if \_\_name\_\_ == '\_\_main\_\_':

main()

**Пример работы программного кода:**

Часть операций проделанных в программе предствлена на рисунке 1.

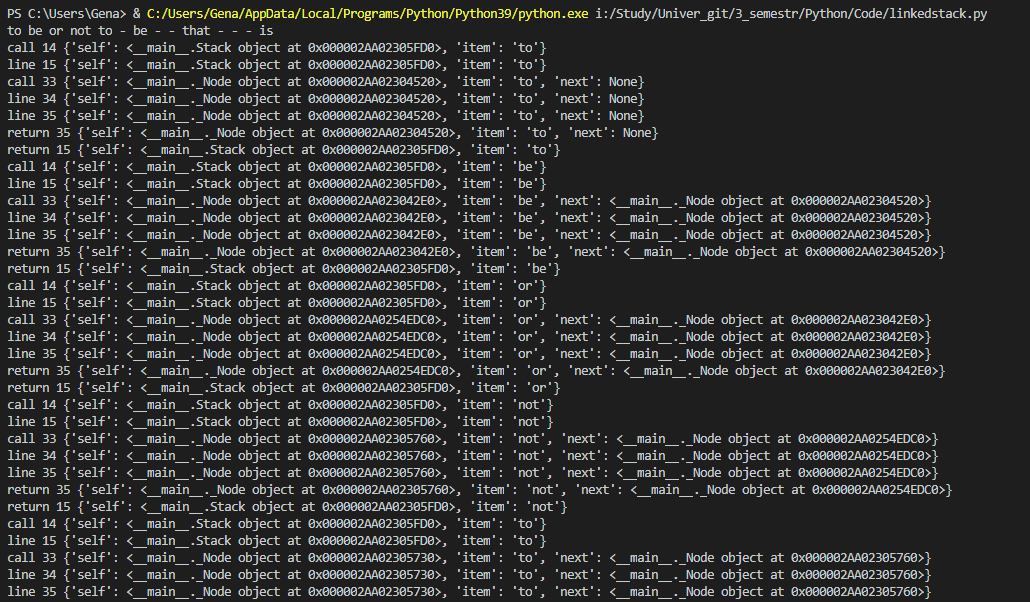


Рисунок 1– Часть traceback работы программы