

CSED232 Object-Oriented Programming

Spring 2019

Programming Assignment #5

Problem Solving

무은재 새내기학부

20180038

박형규

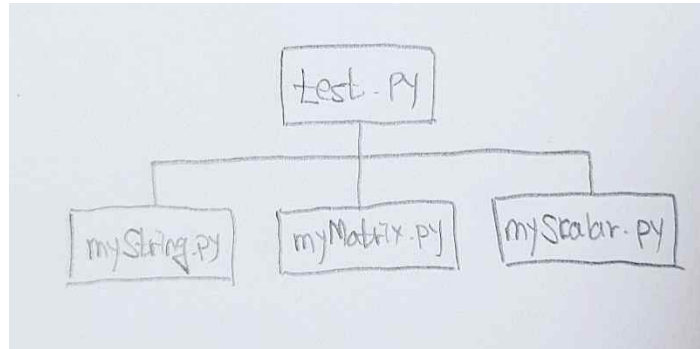
POVIS ID : hyeongkyu

Honor Code : 나는 이 프로그래밍 과제를 다른 사람의 부적절한 도움 없이 완수하였습니다.

1. 프로그램 개요

이번 과제는 python을 이용해서 코딩을 진행하는 것을 목표로 한다. python을 이용하여 class와 operator overloading을 이용한 코딩을 하여 행렬과 문자열을 계산하는 가상의 계산기를 만든다.

2. 전체 구조



이번 프로그램에서는 test.py가 main함수 역할을 하여 실제 사용자가 원하는 작업을 진행시켜준다. myScalar.py, myMatrix.py, myString.py에 존재하는 myScalar, myMatrix, myString class를 test.py에서 import하여 사용한다.

3. Class 설명

myScalar class에서는 `__add__`, `__sub__`, `__mul__`, `__truediv__`의 4가지 operator의 overloading이 이루어진다. 또한 `subMatrix()` 함수는 scalar - matrix 연산을 직접 수행하는 함수이다. operator overloading을 할 때 정해진 type 이외의 type이 들어온다면 `NameError('Not defined')`가 발생된다.

myMatrix class에서는 `__add__`, `__sub__`, `__mul__`, `__truediv__`의 4가지 operator의 overloading이 이루어진다. `addMatrix()` 함수는 matrix + matrix 연산을 직접 수행시켜주는 함수이고, `subScalar()` 함수는 matrix - scalar 연산을 수행시켜주는 함수이고, `subMatrix()` 함수는 matrix - matrix 연산을 수행시켜주는 함수이다. 또한 `mulScalar()` 함수는 matrix * scalar 연산을 수행해주는 함수이고, `mulMatrix()` 함수는 matrix * matrix 연산을 수행해주는 함수이며 `divScalar()` 함수는 matrix / scalar 연산을 수행해주는 함수이다. operator overloading을 할 때 정해진 type 이외의 type이 들어온다면 `NameError('Not defined')`가 발생된다. 또한 `addMatrix()`, `subMatrix()`에서는 두 행렬의 row와 col값이 각각 같지 않으면 `NameError('Dimension mismatch')`를 발생시켰고 `mulMatrix()`에서는 첫 번째 행렬

의 col 값과 두 번째 행렬의 row 값이 같지 않으면 `NameError('Dimension mismatch')`를 발생시켰다.

`myString` class에서는 `__add__`, `__mul__`의 2가지 operator의 overloading이 이루어진다. operator overloading을 할 때 정해진 type 이외의 type이 들어온다면 `NameError('Not defined')`가 발생된다.

4. 프로그램 실행 방법

`test.py` 파일에서 `myScalar`, `myMatrix`, `myString` class를 import하고 원하는 연산을 진행한 후 실행시킨다.

5. 프로그램 실행 예제

```
from myScalar import myScalar
from myMatrix import myMatrix
from myString import myString

# main function
if __name__ == '__main__':
    a = myScalar(3.0)
    aa = myScalar(7.2)
    b = myMatrix(2,3,[0.0,0.0,1.0,0.0,2.0,0.0])
    c = myString('abc')
    d = myMatrix(2,3,[1.0,1.0,2.0,3.0,1.0,0.0])
    dd = myMatrix(3,2,[1.0,1.0,2.0,3.0,1.0,0.0])
    e = myString('efg')
    f = myScalar(-4.0)

    q1 = a + aa
    q2 = a + b
    q3 = a + c
    print(q1, end='\n')
    print(q2, end='\n')
    print(q3, end='\n')

    q4 = a - aa
    q5 = a - b
    print(q4, end='\n')
    print(q5, end='\n')

    q6 = a * aa
    q7 = a * b
    q8 = a * c
    q10 = f * c
    print(q6, end='\n')
    print(q7, end='\n')
    print(q8, end='\n')
    print(q10, end='\n')

    q9 = a / aa
    print(q9, end='\n')

    w1 = b + a
    w2 = b + d
    print(w1, end='\n')
    print(w2, end='\n')

    w3 = b - a
    w4 = b - d
    print(w3, end='\n')
    print(w4, end='\n')

    w5 = b * a
    w6 = b * dd
    print(w5, end='\n')
    print(w6, end='\n')

    w7 = b / a
    print(w7, end='\n')

    e1 = c + a
    e2 = c + e
    print(e1, end='\n')
    print(e2, end='\n')

    e3 = c * a
    e4 = c * f
    print(e3, end='\n')
    print(e4, end='\n')
```

`test.py`에 있는 `main` 함수의 코드가 위 그림과 같을 때 실행되는 결과는 아래와 같다.

```

>>>
RESTART: C:\Users\hyeongkyu\AppData\Local\Programs\Python\Python37-32\test.py
10.2
[[3.0, 3.0, 4.0], [3.0, 5.0, 3.0]]
3.0abc
-4.2
[[3.0, 3.0, 2.0], [3.0, 1.0, 3.0]]
21.6
[[0.0, 0.0, 3.0], [0.0, 6.0, 0.0]]
abcabcabc
cbacbacba
0.4166666666666667
[[3.0, 3.0, 4.0], [3.0, 5.0, 3.0]]
[[1.0, 1.0, 3.0], [3.0, 3.0, 0.0]]
[[-3.0, -3.0, -2.0], [-3.0, -1.0, -3.0]]
[[-1.0, -1.0, -1.0], [-3.0, 1.0, 0.0]]
[[0.0, 0.0, 3.0], [0.0, 6.0, 0.0]]
[[1.0, 0.0], [4.0, 6.0]]
[[0.0, 0.0, 0.3333333333333333], [0.0, 0.6666666666666666, 0.0]]
3.0abc
abcefg
abcabcabc
cbacbacba
>>> |

```

아래 그림은 각각 test.py에 추가한 코드에 따라 Initialization Fail과 Not Defined, Dimension mismatch가 발생하는 모습이다.

```

p1 = myScalar("hi")
#p2 = myString(4.0)
p3 = myMatrix(3,2,[1.0,1.0,2.0,3.0,1.0,0.0])
p4 = myMatrix(2,3,[1.0,1.0,2.0,3.0,1.0,0.0])
p5 = myMatrix(2,3,[0.0,0.0,1.0,0.0,2.0,0.0])

#pp1 = p3 + p4
#pp2 = p3 - p4
#pp3 = p4 * p4

```

```

Traceback (most recent call last):
  File "C:\Users\hyeongkyu\AppData\Local\Programs\Python\Python37-32\test.py", line 68, in <module>
    p1 = myScalar("hi")
  File "C:\Users\hyeongkyu\AppData\Local\Programs\Python\Python37-32\myScalar.py", line 9, in __init__
    raise NameError('Initialization Fail')
NameError: Initialization Fail
>>> |

```

```

#p1 = myScalar("hi")
p2 = myString(4.0)
p3 = myMatrix(3,2,[1.0,1.0,2.0,3.0,1.0,0.0])
p4 = myMatrix(2,3,[1.0,1.0,2.0,3.0,1.0,0.0])
p5 = myMatrix(2,3,[0.0,0.0,1.0,0.0,2.0,0.0])

#pp1 = p3 + p4
#pp2 = p3 - p4
#pp3 = p4 * p4

```

```

Traceback (most recent call last):
  File "C:\Users\hyeongkyu\AppData\Local\Programs\Python\Python37-32\test.py", line 69, in <module>
    p2 = myString(4.0)
  File "C:\Users\hyeongkyu\AppData\Local\Programs\Python\Python37-32\myString.py", line 9, in __init__
    raise NameError('Initialization Fail')
NameError: Initialization Fail
>>> |

```

```

#p1 = myScalar("hi")
#p2 = myString(4.0)
p6 = myMatrix(-1, 3, [1.0,1.0,2.0,3.0,1.0,0.0])
p3 = myMatrix(3,2,[1.0,1.0,2.0,3.0,1.0,0.0])
p4 = myMatrix(2,3,[1.0,1.0,2.0,3.0,1.0,0.0])
p5 = myMatrix(2,3,[0.0,0.0,1.0,0.0,2.0,0.0])

#pp1 = p3 + p4
#pp2 = p3 - p4
#pp3 = p4 * p4

```

```

Traceback (most recent call last):
  File "C:\Users\hyeongkyu\AppData\Local\Programs\Python\Python37-32\test.py", line 70, in <module>
    p6 = myMatrix(-1, 3, [1.0,1.0,2.0,3.0,1.0,0.0])
  File "C:\Users\hyeongkyu\AppData\Local\Programs\Python\Python37-32\myMatrix.py", line 21, in __init__
    raise NameError('Initialization Fail')
NameError: Initialization Fail
>>> |

```

```

#p1 = myScalar("hi")
#p2 = myString(4.0)
#p6 = myMatrix(-1, 3, [1.0,1.0,2.0,3.0,1.0,0.0])
p3 = myMatrix(3,2,[1.0,1.0,2.0,3.0,1.0,0.0])
p4 = myMatrix(2,3,[1.0,1.0,2.0,3.0,1.0,0.0])
p5 = myMatrix(2,3,[0.0,0.0,1.0,0.0,2.0,0.0])

pp1 = p3 + p4
#pp2 = p3 - p4
#pp3 = p4 * p4

```

```

Traceback (most recent call last):
  File "C:\Users\hyeongkyu\AppData\Local\Programs\Python\Python37-32\test.py", line 75, in <module>
    pp1 = p3 + p4
  File "C:\Users\hyeongkyu\AppData\Local\Programs\Python\Python37-32\myMatrix.py", line 56, in __add__
    return self.addMatrix(op)
  File "C:\Users\hyeongkyu\AppData\Local\Programs\Python\Python37-32\myMatrix.py", line 45, in addMatrix
    raise NameError('Dimension mismatch')
NameError: Dimension mismatch
>>> |

```

```

#p1 = myScalar("hi")
#p2 = myString(4.0)
#p6 = myMatrix(-1, 3, [1.0,1.0,2.0,3.0,1.0,0.0])
p3 = myMatrix(3,2,[1.0,1.0,2.0,3.0,1.0,0.0])
p4 = myMatrix(2,3,[1.0,1.0,2.0,3.0,1.0,0.0])
p5 = myMatrix(2,3,[0.0,0.0,1.0,0.0,2.0,0.0])

#pp1 = p3 + p4
pp2 = p3 - p4
#pp3 = p4 * p4

```

```

Traceback (most recent call last):
  File "C:\Users\hyeongkyu\AppData\Local\Programs\Python\Python37-32\test.py", line 76, in <module>
    pp2 = p3 - p4
  File "C:\Users\hyeongkyu\AppData\Local\Programs\Python\Python37-32\myMatrix.py", line 80, in __sub__
    return self.subMatrix(op)
  File "C:\Users\hyeongkyu\AppData\Local\Programs\Python\Python37-32\myMatrix.py", line 69, in subMatrix
    raise NameError('Dimension mismatch')
NameError: Dimension mismatch
>>> |

```

```
#p1 = myScalar("hi")
#p2 = myString(4,0)
#p6 = myMatrix(-1, 3, [1.0,1.0,2.0,3.0,1.0,0.0])
p3 = myMatrix(3,2,[1.0,1.0,2.0,3.0,1.0,0.0])
p4 = myMatrix(2,3,[1.0,1.0,2.0,3.0,1.0,0.0])
p5 = myMatrix(2,3,[0.0,0.0,1.0,0.0,2.0,0.0])

#pp1 = p3 + p4
#pp2 = p3 - p4
pp3 = p4 * p4
```

```
Traceback (most recent call last):
  File "C:\Users\hyeongkyu\AppData\Local\Programs\Python\Python37-32\test.py", line 77, in <module>
    pp3 = p4 * p4
  File "C:\Users\hyeongkyu\AppData\Local\Programs\Python\Python37-32\myMatrix.py", line 110, in __mul__
    return self.mulMatrix(op)
  File "C:\Users\hyeongkyu\AppData\Local\Programs\Python\Python37-32\myMatrix.py", line 94, in mulMatrix
    raise NameError('Dimension mismatch')
NameError: Dimension mismatch
>>> |
```

```
#p1 = myScalar("hi")
#p2 = myString(4,0)
#p6 = myMatrix(-1, 3, [1.0,1.0,2.0,3.0,1.0,0.0])
p3 = myMatrix(3,2,[1.0,1.0,2.0,3.0,1.0,0.0])
p4 = myMatrix(2,3,[1.0,1.0,2.0,3.0,1.0,0.0])
p5 = myMatrix(2,3,[0.0,0.0,1.0,0.0,2.0,0.0])

#pp1 = p3 + p4
#pp2 = p3 - p4
#pp3 = p4 * p4
pp4 = p3 / p3
```

```
Traceback (most recent call last):
  File "C:\Users\hyeongkyu\AppData\Local\Programs\Python\Python37-32\test.py", line 78, in <module>
    pp4 = p3 / p3
  File "C:\Users\hyeongkyu\AppData\Local\Programs\Python\Python37-32\myMatrix.py", line 125, in __truediv__
    raise NameError('Not defined')
NameError: Not defined
>>> |
```

6. 토론, 결론 및 개선방향

이번 Assignment를 통해 처음 접해보는 python을 이용하는 코딩을 직접 진행해 보았다. c++에서는 존재하지 않는 module이나 magic method의 개념을 이용하여 class를 디자인 하고 operator overloading을 직접 진행해 보면서 이론으로 배웠던 내용들을 직접 학습할 수 있는 경험이 되었다. 이번 과제에서는 행렬과 문자열을 계산하는 가상계산기의 data structure class만을 구현하였지만 이를 assn4에서 배웠던 Qt를 이용하여 그래픽까지 적용한다면 실제 컴퓨터에서 사용하는 계산기 까지도 직접 구현해 볼 수 있을 것이라고 생각한다. 또한 행렬과 문자열 이외에도 tree나 링크드 리스트와 같은 여러 가지 data 구조들을 계산해주는 프로그램 또한 구현 가능 할 것이라고 생각한다.

7. 참고문헌

<https://thrillfighter.tistory.com/275>

<https://bewan.tistory.com/62>