



Object-Oriented Programming Report

Assignment 2-2

Professor	Donggyu Sim
Department	Computer engineering
Student ID	2020202031
Name	Jaehyun Kim

Program 1

□ 문제 설명

python 에서 쓰이는 리스트를 구현하는 문제입니다.

arr[][] 과 같이 문자열의 인덱스를 지정해주면 해당 인덱스에 있는 string 이나 array 를 출력하는 문제입니다.

링크드 리스트 형식으로 재귀를 활용하여 각 노드에 string 을 저장했습니다. 만약 list 의 요소가 string 이라면 옆으로 노드를 동적할당 및 연결해서 저장하고 array 라면 아래로 노드를 동적할당 및 연결하여 저장합니다. 제가 구현한 List 클래스는 객체지향프로그래밍실습 ppt 에 나온 그림 예시에서 array 의 끝에 빈 노드가 하나 추가돼있는 상태입니다.

□ 결과 화면

```
[Lorem, ipsum, dolor ,[sit, amet ], vis , antiopam]
arr
[Lorem, ipsum, dolor[sit, amet]vis, antiopam]]
exit
```

string, ',', brackets 사이에 공백이 얼마가 들어가있든 상관 없이 노드에 string 과 array 들이 잘 저장돼있는 것을 확인할 수 있습니다.

```
[[plz, somebody, help , me], [i, am, ,so], tired]
arr
[[plz, somebody, help, me][i, am, so]tired]]
exit
```

brackets 가 연속으로 나오고 array 가 연속으로 나와도 잘 저장됩니다.

❑ 고찰

각 노드클래스에 string 을 저장하는 것은 끝마쳤으나, 시간 부족으로 인하여 사용자로부터 입력 받은 출력 조건에 맞춰 array 를 출력하는 것을 구현하지 못했습니다.

재귀를 이용해서 노드들을 초기화해주는 과정에서 생각지도 못한 예외들이 많이 발생하여 시간이 많이 부족했습니다.

노드들을 출력하는 과정에서 ','를 적당한 위치에 출력하는 것 또한 좋은 조건 값이 생각나지 않아 완벽히 구현하지 못했습니다.

Program 2

❑ 문제 설명

double 형 변수를 private 멤버로 가지는 class A 를 적절한 생성자와 소멸자를 포함하여 실행시키는 문제입니다.

❑ 결과 화면

```
20
```

```
Class A is created with 20
```

```
Class A is destroyed with 20
```

20 을 class A 의 객체 생성자의 매개변수로 전달하여 객체의 멤버변수 num 을 20 으로 초기화하고 동시에 문자열까지 출력되는 것을 확인할 수 있습니다.

main 이 종료되면서 local 로 선언된 객체의 소멸자도 실행이 되면서 문자열이 잘 출력되는 것을 확인할 수 있습니다.

```
5.12987
```

```
Class A is created with 5.12987
```

```
Class A is destroyed with 5.12987
```

5.12987 을 class A 의 객체 생성자의 매개변수로 전달하여 객체의 멤버변수 num 을 5.12987 로 초기화하고 동시에 문자열까지 출력되는 것을 확인할 수 있습니다.

main 이 종료되면서 local 로 선언된 객체의 소멸자도 실행이 되면서 문자열이 잘 출력되는 것을 확인할 수 있습니다.

□ 고찰

사실 객체지향프로그래밍으로서 C++를 배우고 있지만 지금까지의 내용들을 C와 별반 다를 것이 없었습니다. 객체지향프로그래밍의 객체 class 이 아직 익숙하지 않은 현상 상황에서 생성자와 소멸자를 사용하는 예제를 통해 객체의 생성자와 소멸자가 어떻게 작동하는지에 대해 이해할 수 있었습니다. 생성자는 객체 생성과 동시에, 소멸자는 프로그램 종료와 동시에 실행됨을 알게 되었습니다.

Program 3

□ 문제 설명

텍스트파일에 저장 되어있는 개행으로 행을, ','로 열을 구분하는 실수의 모음행렬을 float 형 이차원배열에 저장하고 이를 출력하는 메서드를 만드는 문제입니다.

먼저 행과 열의 개수를 구하기 위해 getline 함수로 문자열을 읽어들이면서 행의 개수를 하나씩 늘려가며 행의 개수를 구했습니다. 마지막 문자열에서 ','의 개수를 구하고 1을 더해 열의 개수도 구했습니다. rows와 cols 크기만큼 data에 이차원 배열을 동적할당했습니다.

wildcard 과제에서 만들어 썼던 split 함수와 stdlib.h 헤더의 atof 함수를 사용했습니다. 다시 처음부터 getline 함수로 문자열을 읽어들이면서 split 메서드로 ','를 기준으로 문자열을 나눠줬습니다. 나눠진 문자열들을 모두 실수의 형태를 띄고 있으므로 atof 메서드로 float 형으로 반환하고 이를 data 이차원배열에 저장했습니다.

소멸자로 data에 할당된 배열을 해제해줬습니다.

□ 결과 화면

```
"CSVLoader_Data.txt" has 97 rows and 2 cols
```

```
6.1101 17.592
5.5277 9.1302
8.5186 13.662
7.32 11.854
5.8598 6.8233
8.3829 11.886
7.4764 4.3483
8.5781 12
6.4862 6.5987
5.546 3.8166
5.7107 3.2522
14.164 15.505
5.734 3.1551
8.4084 7.2258
5.6407 0.71618
5.3794 3.5129
6.3654 5.3048
5.1301 0.56077
6.4296 3.6518
7.708 5.3893
6.1891 3.1386
20.27 21.767
5.4901 4.263
6.3261 5.1875
5.5649 3.825
18.945 22.638
12.828 13.501
10.957 7.467
13.176 14.692
22.203 24.147
5.2524 -1.22
6.5894 5.9966
9.2482 12.134
5.8918 1.8495
8.2111 6.5426
7.9334 4.5623
8.959 4.1164
5.6063 3.3928
12.836 10.117
```

```
6.3534 5.4974
5.4069 0.55657
6.8825 3.9115
11.708 5.3854
5.7737 2.4406
7.8247 6.7318
7.931 1.463
5.702 5.1337
5.8014 1.844
11.7 8.43
5.5416 1.179
7.5402 6.7504
5.3077 1.8396
7.4239 4.2885
7.6031 4.9981
6.3328 1.4233
6.3589 -1.4211
6.2742 2.4756
5.6397 4.6042
9.3102 3.9624
9.4536 5.4141
8.8254 5.1694
5.1793 -0.74279
21.279 17.929
14.908 12.54
18.959 17.54
7.2182 4.8852
8.2951 5.7442
10.236 7.7754
5.4994 1.173
20.341 20.992
10.136 6.6799
7.3345 4.259
6.62 1.2784
7.2259 3.3411
5.269 -2.6807
6.5479 0.29678
7.5386 3.8845
5.365 5.7014
10.274 6.7526
5.1077 2.576
```

```
5.1077 2.576
5.7292 0.47953
5.1884 0.20421
6.3557 0.67861
9.7687 7.5435
6.5159 5.3436
8.5172 4.2415
9.1802 6.7981
6.2 0.92695
5.5204 0.152
5.594 2.8214
5.7077 1.8451
7.6366 4.2959
5.8707 7.2029
5.3054 1.9869
8.2934 0.14454
13.394 9.551
5.4369 0.61705
```

```
"Experiment.txt" has 7 rows and 4 cols
```

```
5.3794 3.5129 6.3654 5.3048
5.1301 0.56077 6.4296 3.6518
7.708 5.3893 6.1891 3.1386
20.27 21.767 5.4901 4.263
6.3261 5.1875 5.5649 3.825
18.945 22.638 12.828 13.501
10.957 7.467 14.555 15.2324
```

첫번째 줄에 rows 와 cols 개수는 getRows 와 getCols 메서드로 반환한 rows 와 cols 를 출력한 것이고 print 로 data 값을 출력했습니다.

□ 고찰

data text 파일 마지막에 개행이 들어가 있어서 ',' 개수로 열의 개수를 구하는 생성자와 parse 메서드에서 열의 개수를 제대로 구하지 못하는 문제점이 발생해서 헤매다가 파일 마지막에 개행 문자를 지우니 프로그램이 제대로 작동했습니다.

parse 메서드는 결국 파일명을 인자로 받는 생성자와 하는 역할이 같기 때문에 생성자의 코드를 그대로 복사했었는데, 생성자로 data 를 채운 상태에서 parse 를 실행시키면 본래 data 에 저장돼있던 heap 데이터가 해제되지 않고 그대로 남아있는 문제점을 발견하고 parse 메서드가 실행되면 메모리를 해제하도록 코드를 수정했습니다.

Program 4

□ 문제 설명

Clock class 를 생성하고 Clock 의 초기시간을 설정해줄 set 메서드들과 1 단위로 시간, 분, 초를 증가시켜줄 increase 메서드들을 정의해주는 문제입니다.

□ 결과 화면

```
Clock turns on
Set beggining time
Enter seconds: 92200
It's 1:36:40 now
input among H, M, S, X: H
It's 2:36:40 now
input among H, M, S, X: H
It's 3:36:40 now
input among H, M, S, X: H
It's 4:36:40 now
input among H, M, S, X: M
It's 4:37:40 now
input among H, M, S, X: M
It's 4:38:40 now
input among H, M, S, X: M
It's 4:39:40 now
input among H, M, S, X: A
unvalid input!
input among H, M, S, X: A
unvalid input!
input among H, M, S, X: M
It's 4:40:40 now
input among H, M, S, X: S
It's 4:40:41 now
input among H, M, S, X: S
It's 4:40:42 now
input among H, M, S, X: S
It's 4:40:43 now
input among H, M, S, X: X
It's 4:40:43 now
```

92200 을 입력하자 setTime 메소드가 hour 를 1, minute 을 36, second 를 40 으로 초기화하고 시각을 출력합니다. 사용자로부터 'H'를 입력 받으면 increaseHour()메소드가, 'M'을 입력 받으면 increaseMinute()메소드가, 'S'를 입력 받으면 increaseSecond()메소드가 실행되어 hour, minute, second 가 1 씩 증가하는 것을 확인할 수 있습니다.

사용자로부터 'X'를 입력 받으면 프로그램이 종료됩니다.

```
Clock turns on
Set beggining time
Enter seconds: 86399
It's 23:59:59 now
input among H, M, S, X: H
It's 0:59:59 now
input among H, M, S, X: M
It's 1:0:59 now
input among H, M, S, X: S
It's 1:1:0 now
input among H, M, S, X: X
It's 1:1:0 now
```

```
Clock turns on
Set beggining time
Enter seconds: 86399
It's 23:59:59 now
input among H, M, S, X: S
It's 0:0:0 now
input among H, M, S, X: X
It's 0:0:0 now
```

다음과 같이 hour, minute, second 중 하나가 범위를 벗어나는 경우에 대한 예외처리로 잘 실행됨을 확인할 수 있습니다.

□ 고찰

increaseMinute()과 increaseSecond() 메소드에서 minute 과 second 가 60 이상으로 넘어가면 0 으로 초기화하고 hour 와 minute 을 1 씩 증가시켜줘야 합니다. 하지만 minute 과 second 가 60 이상이 됐을 때 hour 와 minute 이 1 이 증가하므로써 hour 와 minute 또한 범위를 넘어가게 될 경우를 고려하지 못해 에러가 발생했습니다. 이를 minute 과 second 가 60 이상이 됐을 때 0 으로 초기화해줌과 동시에 increaseHour(), increaseMinute()메소드를 호출하는 것으로 간단히 해결했습니다.