

11624_W12_2022076762_변경민

P1. merge 함수 구현

```
In [33]: import random
import time

l = list(range(10))
random.shuffle(l)
print(l)

def merge(left, right):
    merged = list()
    while( len(left)> 0) or (len(right) > 0):
        if len(right) <= 0:
            print('merged -->', merged+left)
            return merged + left
        elif len(left) <= 0:
            print('merged -->', merged+right)
            return merged + right
        else:
            if left[0] <= right[0]:
                merged.append(left[0])
                del left[0]
            else:
                merged.append(right[0])
                del right[0]
    print('merged -->', merged)
    return merged

def selectSort(l):
```

```

def selectSort(l):
    for i in range(len(l)-1):
        min_idx = i
        for j in range(i+1, len(l)):
            if l[min_idx] > l[j]:
                min_idx = j
        tmp = l[i]
        l[i] = l[min_idx]
        l[min_idx] = tmp
    return l

print(merge(selectSort(l[:5]), selectSort(l[5:])))

```

[3, 8, 4, 6, 7, 1, 9, 2, 5, 0]
merged --> [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]

P2. 합병정렬 구현

```

In [14]: import random
import time

l = list(range(10))
random.shuffle(l)
print(l)

def mergeSort(l):
    if len(l) <= 1:
        return l
    mid = len(l)//2
    left = mergeSort(l[:mid])
    right = mergeSort(l[mid:])
    return merge(left, right)

def merge(left, right):

```

```

def merge(left, right):
    merged = list()
    while( len(left)> 0) or (len(right) > 0):
        if len(right) <= 0:
            print('merged -->', merged+left)
            return merged + left
        elif len(left) <= 0:
            print('merged -->', merged+right)
            return merged + right
        else:
            if left[0] <= right[0]:
                merged.append(left[0])
                del left[0]
            else:
                merged.append(right[0])
                del right[0]
    print('merged -->', merged)
    return merged

```

```

start = time.time()
sorted_l = mergeSort(l)
print(sorted_l)
print('정렬에 걸린시간(s):', time.time()-start)

```

```

[7, 4, 8, 0, 5, 9, 2, 6, 3, 1]
merged --> [4, 7]
merged --> [0, 5]
merged --> [0, 5, 8]
merged --> [0, 4, 5, 7, 8]
merged --> [2, 9]
merged --> [1, 3]
merged --> [1, 3, 6]
merged --> [1, 2, 3, 6, 9]
merged --> [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
정렬에 걸린시간(s): 0.0005080699920654297

```

P3. 수행시간 비교

	n	100	500	1,000	5,000	10,000	20,000	40,000	80,000
선택정렬	0.0	0.0059	0.0229	0.5764	2.3068	9.3177	38.9889	158.5231	
삽입정렬	0.0	0.0059	0.02984	0.6896	2.8675	11.3276	45.4666	176.6669	
합병정렬	0.0	0.0023	0.0044	0.0286	0.0533	0.1168	0.2958	0.8976	

```
In [10]: import random
import time

def sortNList(n):
    l = list(range(n))
    random.shuffle(l)
    start = time.time()
    sorted_l = mergeSort(l)
    print(n, '개 리스트 정렬에 걸린시간(s):', time.time()-start)

def mergeSort(l):
    if len(l) <= 1:
        return l
    mid = len(l)//2
    left = mergeSort(l[:mid])
    right = mergeSort(l[mid:])
    return merge(left, right)

def merge(left, right):
    merged = list()
    while( len(left)> 0) or (len(right) > 0):
        if len(right) <= 0:
            return merged + left
        if len(left) <= 0:
            return merged + right
        if left[0] < right[0]:
            merged.append(left.pop(0))
        else:
            merged.append(right.pop(0))
    return merged
```

```
        elif len(left) <= 0:
            return merged + right
        else:
            if left[0] <= right[0]:
                merged.append(left[0])
                del left[0]
            else:
                merged.append(right[0])
                del right[0]
    return merged

sortNList(100)
sortNList(500)
sortNList(1000)
sortNList(5000)
sortNList(10000)
sortNList(20000)
sortNList(40000)
sortNList(80000)
```

100 개 리스트 정렬에 걸린시간(s): 0.000621795654296875
500 개 리스트 정렬에 걸린시간(s): 0.0023386478424072266
1000 개 리스트 정렬에 걸린시간(s): 0.004401206970214844
5000 개 리스트 정렬에 걸린시간(s): 0.028678178787231445
10000 개 리스트 정렬에 걸린시간(s): 0.05339479446411133
20000 개 리스트 정렬에 걸린시간(s): 0.11685609817504883
40000 개 리스트 정렬에 걸린시간(s): 0.2958838939666748
80000 개 리스트 정렬에 걸린시간(s): 0.8976931571960449

In []:

