

알고리즘 1주차 실습

202001258 홍석준

1. 에라토스테네스의 체

ADL:

```
eratos(a, n)
    a[1] ← 0;
    i ← 2;
    while (i < n/2) do {
        j ← 2;
        while (i*j ≤ n) do {
            a[i*j] ← 0;
            j ← j+1;
        }
        i ← i+1;
    }
    return a;
end eratos()
```

파이썬 코드:

```
def eratos(a, n):
    a[1] = 0
    i = 2
    while(i < n/2):
        j = 2
        while(i*j <= n):
            a[i*j] = 0
            j = j+1
        i = i+1
    return a

N = int(input('N = '))
while N < 1:
    print(N, '은(는) 자연수가 아닙니다.')
    N = int(input('N = '))
```

```
A = list(range(N+1))
res = eratos(A, N)
for i in range(N+1):
    if res[i]: print(i, end=' ')
```

실행 결과:

```
PS C:\Users\HONG\Desktop\홍석순\인천대학교\24-2/알고리즘/2. 실습/1주차/eratos.py"
N = 50
2 3 5 7 11 13 17 19 23 29 31 37 41 43 47
```

2. 두 수의 합

ADL:

twoSum(a,n,t)

$i \leftarrow 0$;

while($i < n$) do{

 while ($i < n$) do{

 if($a[i] + a[j] = t$) then

 print($i+1$ 번째 원소와 $j+1$ 번째 원소)

$j \leftarrow j+1$;

$i \leftarrow i+1$;

 }

 }

end twoSum()

파이썬 코드:

```
def twoSum(a, n, t):
    i=0
    while(i<n):
        j=i+1
        while(j<n):
            if((a[i]+a[j])==t):
                print("{}번째와 {}번째의 원소 ".format(i+1,j+1))
            j=j+1
        i=i+1
```

```

import random
N =int(input('리스트의 원소 개수 : '))
A =[]
for i in range(N):
    A.append(random.randint(1, N*2))
print('리스트 : ', A)
T =int(input('목표값 입력 : '))
print('두 수의 합이 %d인 원소 쌍'%T)
twoSum(A, N, T)

```

실행 결과:

```

-2\알고리즘\2. 실습\1주차\twoSum.py" 4-2\알고리즘\2. 실습>
리스트의 원소 개수 : 10
리스트 : [10, 3, 18, 14, 3, 14, 13, 6, 9, 16]
목표값 입력 : 24
두 수의 합이 24인 원소 쌍
1번째와 4번째의 원소
1번째와 6번째의 원소
3번째와 8번째의 원소
PS C:\Users\HONG\Desktop\홍석준\인천대학교\24-2\알고리즘\2. 실습>

```

3. 파스칼의 수 삼각형

ADL:

```

pascalTriangle(h)
    res←공백리스트;
    pre←공백리스트;
    pre에 1 추가;
    res에 pre 추가;
    i←1;
    while (i<h) do {
        cur ← 공백리스트;
        cur에 1 추가;
        j←0;

```

```

        while(j<pre의 길이 -1) do {
            cur에 pre[j] 와 pre[j+1]의 합 추가;
            j←j+1;
        }
        cur에 1 추가;
        res에 cur 추가;
        pre←cur;
        i←i+1;
    }
    return res;
end pascalTriangle()

```

파이썬 코드:

```

def pascalTriangle(h):
    res = []
    pre = []
    pre.append(1)
    res.append(pre)
    i=1
    while(i<h):
        cur = []
        cur.append(1)
        j=0
        while(j<(len(pre)-1)):
            cur.append((pre[j]+pre[j+1]))
            j=j+1
        cur.append(1)
        res.append(cur)
        pre=cur
        i=i+1
    return res

H =int(input('높이 입력 : '))
result =pascalTriangle(H)

```

```

for i in range(len(result)):
    for j in range(len(result[i])):
        print(result[i][j], end=' ')
    print()

```

실행 결과:

```

-2/알고리즘/2. 실습/1주차/pascalTriangle.py"

```

높이 입력 : 10

```

1
1 1
1 2 1
1 3 3 1
1 4 6 4 1
1 5 10 10 5 1
1 6 15 20 15 6 1
1 7 21 35 35 21 7 1
1 8 28 56 70 56 28 8 1
1 9 36 84 126 126 84 36 9 1

```

PS C:\Users\HONG\Desktop\홍석준\인천대학교\24-2\알고리즘\2. 실습> █

4. 최장 연속 순차

ADL:

longSequence(a,s,n)

max_len←1;

```

i←0;
while(i<n) do {
    left←a[i]-1;
    right←a[i]+1;
    count←1;
    while(left∈s) do {
        count←count+1;
        집합 s에서 left 제거;
        left←left-1;
    }
    while(right∈s) do {
        count←count+1;
        집합 s에서 right 제거;
        right←right+1;
    }
    max_len←max(max_len,count);
    i←i+1;
}
return max_len;
end longestSequence()

```

파이썬 코드:

```

def longestSequence(a, s, n):
    max_len=1
    i=0
    while(i<n):
        left=a[i]-1
        right=a[i]+1
        count=1
        while(left in s):
            count=count+1
            s.remove(left)
            left=left-1
        while(right in s):
            count=count+1
            s.remove(right)
            right=right+1
    
```

```

        max_len=max(max_len,count)
        i=i+1
    return max_len

import random
N =int(input('난수의 개수 : '))
while N <1:
    print('난수의 개수는 자연수여야 합니다.')
    N =int(input('난수의 개수 : '))
A =[]
for i in range(N):
    A.append(random.randint(1, N))
print('리스트 : ', A)
S =set(A)
print('집합 : ', S)
print('최장 연속 순차의 길이 : ', longestSequence(A, S, N))

```

실행 결과:

```

난수의 개수 : 10
리스트 : [1, 5, 5, 2, 8, 3, 8, 1, 7, 4]
집합 : {1, 2, 3, 4, 5, 7, 8}
최장 연속 순차의 길이 : 5
PS C:\Users\HONG\Desktop\홍석준\인천대학교\24-2\알고리즘\2. 실습> 

```

4-1

ADL:

longSequence(a,s,n)

```

    max_len←1;
    i←0;
    longest←공백리스트;
    while(i<n) do {
        now←공백리스트;
        left←a[i]-1;
        right←a[i]+1;
        count←1;
        now에 a[i] 추가;
        while(left∈s) do {

```

```

        count←count+1;
        now에 left 추가;
        집합 s에서 left 제거;
        left←left-1;
    }
    while(right∈s) do {
        count←count+1;
        now에 right 추가;
        집합 s에서 right 제거;
        right←right+1;
    }
    max_len←max(max_len,count);
    if (longest의 길이 ≤ now의 길이) then
        longest ← now;
    i←i+1;
}
longest 오름차순 정렬;
return max_len, longest;
end longestSequenceList()

```

파이썬 코드:

```

def longestSequenceList(a, s, n):
    max_len=1
    i=0
    longest=[]
    while(i<n):
        now=[]
        left=a[i]-1
        right=a[i]+1
        count=1
        now.append(a[i])
        while(left in s):
            count=count+1
            now.append(left)
            s.remove(left)
            left=left-1
        while(right in s):
            count=count+1
            now.append(right)

```



```

        s.remove(right)
        right=right+1
        max_len=max(max_len,count)
        if(len(longest)<=len(now)):
            longest =now
        i=i+1
    longest.sort()
    return max_len,longest

import random
N =int(input('난수의 개수 : '))
while N <1:
    print('난수의 개수는 자연수여야 합니다.')
    N =int(input('난수의 개수 : '))
A =[]
for i in range(N):
    A.append(random.randint(1, N))
print('리스트 : ', A)
S =set(A)
print('집합 : ', S)
len_longest, longest_list =longestSequenceList(A, S, N)
print('최장 연속 순차의 길이 : ', len_longest)
print('최장 연속 순차 : ', longest_list)

```

실행 결과:

```

-27\알고리즘\2. 실습\1주차\longestSequenceList.py
난수의 개수 : 10
리스트 :  [10, 9, 3, 6, 6, 3, 5, 10, 1, 7]
집합 :  {1, 3, 5, 6, 7, 9, 10}
최장 연속 순차의 길이 :  3
최장 연속 순차 :  [5, 6, 7]
PS C:\Users\HONG\Desktop\홍석준\인천대학교\24-2\알고리즘\2. 실습> █

```