

HW2 – Prime

(정수론)

18013195

정보보호학과

이풍원

1) `is_prime(n)`: it returns 1 if `n` is prime and return 0 otherwise. To test wheter it is prime or not, you can use a trivial division.

```
2 def is_prime(n):
3     for i in range(2,int(n**1/2)+1):
4         if n%i==0:
5             return 0
6     return 1
```

3 행: 2 부터 인자로 받은 `n` 값의 제곱근만큼 반복한다.

4~5 행: `n` 에 `i` 를 나눠 나머지 값이 0 이면 소수가 아니기 때문에 0 을 반환한다.

6 행: 위의 for 문에서 조건을 만족하지 않아 탈출하면 소수이기 때문에 1 을 반환한다.

2) `generate_all_primes(n)`: it returns a list that contains all primes from 2 to `n`.

```
7 def generate_all_primes(n):
8     prime_list=[]
9     for i in range(2,n+1):
10         if is_prime(i):
11             prime_list.append(i)
12     return prime_list
```

8 행: 소수들을 담을 `prime_list` 를 list 형으로 선언한다.

9 행: 2 부터 `n` 까지 소수 판별을 하고 그만큼 반복한다.

10~11 행: if 문에 `is_prime` 함수를 넣어 `i` 를 넣고 소수 판별을 하고 `is_prime` 은 소수이면 1 을 반환하기 때문에 참이며 `prime_list` 에 `i` 를 추가한다.

12 행: 소수들이 담긴 `prume_list` 를 반환한다.

3) `generate_random_prime(a, b)`: it return a random prime that is chosen uniformly from a to b.

```
14 def generate_random_prime(a,b):
15     while True:
16         r_num=random.randrange(a, b+1)
17         if is_prime(r_num):
18             return r_num
```

15 행: 소수를 찾을 때 까지 반복하기 위해 True 를 넣어 무한 루프 한다.

16 행: 랜덤 함수인 `random.randrange()` 를 이용하여 a 부터 b 까지의 랜덤 값을 발생시켜 `r_num` 에 저장한다.

17~18 행: `is_prime` 함수를 이용해 `r_num` 를 넣어 참이면 1 를 반환하기 때문에 만족하면 소수 `r_num` 을 반환한다.

4) 실행 결과

```
is_prime(11)= 1
is_prime(253)= 0
is_prime(65537)= 1
generate_all_primes(50)= [2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47]
generate_all_primes(100)= [2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97]
generate_all_primes(1000)= [2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, 101, 103, 107, 109, 113, 127, 131, 137, 139, 149, 151, 157, 163, 167, 173, 179, 181, 191, 193, 197, 199, 211, 223, 227, 229, 233, 239, 241, 251, 257, 263, 269, 271, 277, 281, 283, 293, 307, 311, 313, 317, 331, 337, 347, 349, 353, 359, 367, 373, 379, 383, 389, 397, 401, 409, 419, 421, 431, 433, 439, 443, 449, 457, 461, 463, 467, 479, 487, 491, 499, 503, 509, 521, 523, 541, 547, 557, 563, 569, 571, 577, 587, 593, 599, 601, 607, 613, 617, 619, 631, 641, 643, 647, 653, 659, 661, 673, 677, 683, 691, 701, 709, 719, 727, 733, 739, 743, 751, 757, 761, 769, 773, 787, 797, 809, 811, 821, 823, 827, 829, 839, 853, 857, 859, 863, 877, 881, 883, 887, 907, 911, 919, 929, 937, 941, 947, 953, 967, 971, 977, 983, 991, 997]
generate_random_prime(2,11)= 5
generate_random_prime(100,200)= 179
generate_random_prime(1000,2000)= 1543
PS C:\Users\vnddn\OneDrive\바탕 화면\js> []
```

5)소스 코드

```
import random
def is_prime(n):
    for i in range(2,int(n**1/2)+1):
        if n%i==0:
            return 0
    return 1
def generate_all_primes(n):
    prime_list=[]
    for i in range(2,n+1):
        if is_prime(i):
            prime_list.append(i)
    return prime_list

def generate_random_prime(a,b):
    while True:
        r_num=random.randrange(a, b+1)
        if is_prime(r_num):
            return r_num

print("is_prime(11)=",is_prime(11))
print("is_prime(253)=",is_prime(253))
print("is_prime(65537)=",is_prime(65537))

print("generate_all_primes (50)=",generate_all_primes (50))
print("generate_all_primes (100)=",generate_all_primes (100))
print("generate_all_primes (1000)=",generate_all_primes (1000))

print("generate_random_prime(2,11)=",generate_random_prime (2, 11))
print("generate_random_prime(100,200)=",generate_random_prime (100, 200))
print("generate_random_prime(1000,2000)=",generate_random_prime (1000, 2000))
```