



코딩과제 #39 (배점: 40점)

1. 설명:

섹시 소수는 $(p, p+6)$ 가 모두 소수인 쌍 (즉, 6씩 차이나는 소수)를 말한다. 예를 들면 5와 11, 7과 13, 461과 467이 섹시 소수의 좋은 예이다.

이번 프로그램 과제는 먼저, 1부터 50000 범위 내의 수 중에서 소수를 찾아내고, 다음으로 이 범위 내에 존재하는 모든 섹시 소수를 찾는 것이 목적이다. 주어진 범위 내에서 소수를 찾기 위한 알고리즘은 에라토스테네스의 체 방법(the Sieve of Eratosthenes)이다. 이 알고리즘은 아래에서 자세히 설명한다.

아래 main 함수를 이용하여 프로그램 숙제를 할 수 있다. 미리 만들어진 main 함수를 이용하여 이 함수 안에서 호출되는 총 네 개의 사용자 정의함수를 직접 구현하는 것이 이번 숙제의 핵심이다. 미리 구현된 main 함수는 아래와 같다.

```
#define HOWMANY 50000 // The range of numbers

void processSieve(int[]);
int getLower();
int getUpper(int);
void showPrimes(int[], int, int);

int main()
{
    int sieve[HOWMANY+1];           // The array of true/false
    int lower = 1, upper = HOWMANY; // Setting initial boundaries

    // the following function call will implement the Sieve algorithm
    processSieve(sieve);

    // the following function call gets the lower boundary for printing
    lower = getLower();

    // the following function call gets the upper boundary for printing
    upper = getUpper(lower);

    // the following function call prints sexy pairs in the lower-upper range
    showPrimes(sieve, lower, upper);
}

// implement processSieve function here

// implement getLower function here

// implement getUpper function here

// implement showPrimes function here
```

※ 유의사항: 본 코딩과제는 반드시 함수를 만들어야 함.



■ 소수를 찾아내는 알고리즘 (The Sieve of Eratosthenes)

어떤 수가 소수인지 아닌지를 알기 위해서는 그 수의 약수가 1과 그 수 자신뿐인지를 검사해야 한다. 주어진 수가 크지 않다면 그 계산식이 복잡하지 않겠지만, 임의의 어떤 수가 주어지고 그 수가 소수인지 아닌지를 판별하는 것은 그렇게 간단한 작업이 아니다. 고대 그리스 수학자 에라토스테네스는 소수를 쉽게 찾는 알고리즘을 개발하였다. 이 알고리즘을 이용하여 processSieve 함수를 구현할 수 있다. 여러분의 이해를 돕기 위해서 아래 그림에서 1부터 19까지의 범위 내의 수 중에서 소수를 찾는 과정을 에라토스테네스의 체 방법을 적용하여 나타냈다.

1) 초기 배열의 모습 (이 배열에서의 인덱스는 각각의 숫자를 나타내고 배열의 값이 1인 경우는 그 수가 소수임을 그렇지 않으면 소수가 아님을 나타낸다):

[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]	[10]	[11]	[12]	[13]	[14]	[15]	[16]	[17]	[18]	[19]
0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

2) 숫자2에 대해 에라토스테네스의 체 방법을 적용한 모습:

[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]	[10]	[11]	[12]	[13]	[14]	[15]	[16]	[17]	[18]	[19]
0	0	1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

3) 숫자3에 대해 에라토스테네스의 체 방법을 적용한 모습::

[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]	[10]	[11]	[12]	[13]	[14]	[15]	[16]	[17]	[18]	[19]
0	0	1	1	0	1	0	1	0	0	0	1	0	1	0	0	0	1	0	1

이러한 과정을 거친 후 현재 배열의 인덱스 2, 3, 5, 7, 11, 13, 17, 19만이 1의 값을 가지고 있음을 알 수 있고, 또한 이 숫자들이 소수임을 나타내고 있다.

2. 실행결과 예시:

출력 예시#1

```
Please enter the lower boundary (between 1 and 50000): 0
Please enter the lower boundary (between 1 and 50000): 50001
Please enter the lower boundary (between 1 and 50000): 150

Please enter the upper boundary (between 150 and 50000): 0
Please enter the upper boundary (between 150 and 50000): 50001
Please enter the upper boundary (between 150 and 50000): 100
Please enter the upper boundary (between 150 and 50000): 200
```

```
Here are all of the sexy prime pairs in the range 150 to 200, one pair per line:
151 and 157
157 and 163
167 and 173
173 and 179
191 and 197
193 and 199
```

There were 6 sexy prime pairs displayed.



출력 예시#2

Please enter the lower boundary (between 1 and 50000): 1

Please enter the upper boundary (between 1 and 50000): 42

Here are all of the sexy prime pairs in the range 1 to 42, one pair per line:

5 and 11
7 and 13
11 and 17
13 and 19
17 and 23
23 and 29
31 and 37

There were 7 sexy prime pairs displayed.

출력 예시#3

Please enter the lower boundary (between 1 and 50000): 49500

Please enter the upper boundary (between 49500 and 50000): 49993

Here are all of the sexy prime pairs in the range 49500 to 49993, one pair per line:

49523 and 49529
49531 and 49537
49597 and 49603
49627 and 49633
49633 and 49639
49663 and 49669
49741 and 49747
49783 and 49789
49801 and 49807
49871 and 49877
49921 and 49927
49937 and 49943

There were 12 sexy prime pairs displayed.