[HW#4] 다중 쓰레드를 통한 병행 작업

```c
1    #include <stdio.h>
2    #include <stdlib.h>
3    #include <unistd.h>
4    #include <pthread.h>
5    #include<semaphore.h>
6    int isPrimeNumber(int n);
7    void* testthread(void* arr);
8    static sem_t semA;
9
```

코드 사용시 필요한 header와

함수 isPrimerNumber와

Thread에서 사용 할

testthread 그리고

Semaphore 변수를 미리 호출

필요한 변수들을 선언 후, n을 쓰레드의 개수, a를 범위에 설정 후 malloc을 통해 pthread 로 실행 할 쓰레드를 배열로 생성

그 후 i번째 쓰레드에서 실행 할 범위를 arr[i][0] 부터 arr[i][1]까지 선언

```c
10   int main(int argc, char* argv[]) {
11       sem_init(&semA, 0, 1);
12       int i, n, a, total = 0;
13       void *tret = NULL;
14       n = atoi(argv[2]); //쓰레드의 개수 n에 저장
15       a = atoi(argv[1]); //범위 a에 저장
16       pthread_t* tid;
17       tid = (pthread_t*)calloc(n, sizeof(pthread_t));//쓰레드의 개수만큼 tid malloc
18       int arr[n][2];
19       for (i = 0; i < n; i++) { //thread n개 생성
20           arr[i][0] = i * a / n + 1;//쓰레드 시작 수
21           arr[i][1] = (i + 1) * a / n;//쓰레드의 끝 수
22       }
```

Sem_wait를 사용해 pthread가 i 를 할당 받기 전에 일단 lock한다. (실행시 unlock)

다음 pthread_join을 통해 thread를 끝내고 받은 return 값을 더해 줘서 total에 더해준 후

Main 이 카운트 값과 자신의 아이디 출력

```c
23       sem_wait(&semA);//i 를 제공 하기 전 한명만 진입하게 block
24       for (i = 0; i < n; i++) {
25           tid[i] = pthread_self();
26           pthread_create(&tid[i], NULL, testthread, (void*)&arr[i]);//thread 생성
27       }
28
29       for (i = 0; i < n; i++) {
30           pthread_join(tid[i], &tret);//리턴받은 k를 주소값에 저장
31           total += *((int*)tret);//int형으로 변환 후 total 에 더해준다
32       }
33
34       printf("The main thread[%ld] has found %d prime numbers in (%d ~ %d).\n",
35           pthread_self(),total,1,a);
36       free(tid);//malloc 해제
37       return 0;
38   }
```

소수인지 확인을 해주는 isPrimeNumber 함수

```c
39  int isPrimeNumber(int n) {
40      int i = 0;
41      int last = n / 2;
42      if (n <= 1)
43          return 0;
44      for (i = 2; i <= last; i++)
45          if ((n % i) == 0)
46              return 0;
47      }
48      return 1;
49  }
```

pthread를 생성 시에 사용하는 testthread함수

시작과 동시에 i를 할당 받았으니 sem_post를 통해 semaphore값을 증가 해 unlock 해준다.

Return 받을 k 의 값은 힙 공간에 저장 해 메인에서 쓸 수 있게 사용

소수 계산 후 pthread_exit 에서 k의 포인터 값을 전달

```c
50  void* testthread(void* arr) {
51      sem_post(&semA);//i를 할당 받고 다시 unblock 해준다
52      int arrint[2];
53      int *k = (int*)malloc(sizeof(int)*1);
54      //메인 쓰레드에서 받을 때 힙 공간에 저장 해야 전달이 되므로 malloc 사용
55      arrint[0] = ((int*)arr)[0];
56      arrint[1] = ((int*)arr)[1];
57      for (int i = arrint[0]; i < arrint[1]+1; i++) {
58          if (isPrimeNumber(i) == 1) {
59              k[0] += 1;
60          }
61      }
62      fprintf(stderr, "Thread[%ld] has found %d prime numbers in (%d ~ %d).\n",
63          pthread_self(), k[0], arrint[0], arrint[1]);
64      pthread_exit(k);//포인터로 전달
65      free(k);
66  }
```

범위

1~

10000

```
yeonggi@DESKTOP-QK10V32:~/system/hw4$ ./hw4_201713025.out 10000 1
Thread[140008588285696] has found 1229 prime numbers in (1 ~ 10000).
The main thread[140008588289856] has found 1229 prime numbers in (1 ~ 10000).
yeonggi@DESKTOP-QK10V32:~/system/hw4$ ./hw4_201713025.out 10000 2
Thread[140431657899776] has found 669 prime numbers in (1 ~ 5000).
Thread[140431649507072] has found 560 prime numbers in (5001 ~ 10000).
The main thread[140431657903936] has found 1229 prime numbers in (1 ~ 10000).
yeonggi@DESKTOP-QK10V32:~/system/hw4$ ./hw4_201713025.out 10000 3
Thread[140265383536384] has found 470 prime numbers in (1 ~ 3333).
Thread[140265375143680] has found 389 prime numbers in (3334 ~ 6666).
Thread[140265366750976] has found 370 prime numbers in (6667 ~ 10000).
The main thread[140265383540544] has found 1229 prime numbers in (1 ~ 10000).
yeonggi@DESKTOP-QK10V32:~/system/hw4$ ./hw4_201713025.out 10000 4
Thread[140386092795648] has found 367 prime numbers in (1 ~ 2500).
Thread[140386084402944] has found 302 prime numbers in (2501 ~ 5000).
Thread[140386076010240] has found 281 prime numbers in (5001 ~ 7500).
Thread[140386067617536] has found 279 prime numbers in (7501 ~ 10000).
The main thread[140386092799808] has found 1229 prime numbers in (1 ~ 10000).
yeonggi@DESKTOP-QK10V32:~/system/hw4$
```

범위

1~

100000

```
yeonggi@DESKTOP-QK10V32:~/system/hw4$ ./hw4_201713025.out 100000 1
Thread[140587014166272] has found 9592 prime numbers in (1 ~ 100000).
The main thread[140587014170432] has found 9592 prime numbers in (1 ~ 100000).
yeonggi@DESKTOP-QK10V32:~/system/hw4$ ./hw4_201713025.out 100000 2
Thread[140617619461888] has found 5133 prime numbers in (1 ~ 50000).
Thread[140617611069184] has found 4459 prime numbers in (50001 ~ 100000).
The main thread[140617619466048] has found 9592 prime numbers in (1 ~ 100000).
yeonggi@DESKTOP-QK10V32:~/system/hw4$ ./hw4_201713025.out 100000 3
Thread[140389530957568] has found 3569 prime numbers in (1 ~ 33333).
Thread[140389522564864] has found 3076 prime numbers in (33334 ~ 66666).
Thread[140389514172160] has found 2947 prime numbers in (66667 ~ 100000).
The main thread[140389530961728] has found 9592 prime numbers in (1 ~ 100000).
yeonggi@DESKTOP-QK10V32:~/system/hw4$ ./hw4_201713025.out 100000 4
Thread[140449137350400] has found 2762 prime numbers in (1 ~ 25000).
Thread[140449128957696] has found 2371 prime numbers in (25001 ~ 50000).
Thread[140449120564992] has found 2260 prime numbers in (50001 ~ 75000).
Thread[140449112172288] has found 2199 prime numbers in (75001 ~ 100000).
The main thread[140449137354560] has found 9592 prime numbers in (1 ~ 100000).
yeonggi@DESKTOP-QK10V32:~/system/hw4$
```

범위

1~

1000000

```
yeonggi@DESKTOP-QK10V32:~/system/hw4$ ./hw4_201713025.out 1000000 1
Thread[139705405859584] has found 78498 prime numbers in (1 ~ 1000000).
The main thread[139705405863744] has found 78498 prime numbers in (1 ~ 1000000).
yeonggi@DESKTOP-QK10V32:~/system/hw4$ ./hw4_201713025.out 1000000 2
Thread[140518215882496] has found 41538 prime numbers in (1 ~ 500000).
Thread[140518207489792] has found 36960 prime numbers in (500001 ~ 1000000).
The main thread[140518215886656] has found 78498 prime numbers in (1 ~ 1000000).
yeonggi@DESKTOP-QK10V32:~/system/hw4$ ./hw4_201713025.out 1000000 3
Thread[139858745657088] has found 28665 prime numbers in (1 ~ 333333).
Thread[139858737264384] has found 25404 prime numbers in (333334 ~ 666666).
Thread[139858728871680] has found 24429 prime numbers in (666667 ~ 1000000).
The main thread[139858745661248] has found 78498 prime numbers in (1 ~ 1000000).
yeonggi@DESKTOP-QK10V32:~/system/hw4$ ./hw4_201713025.out 1000000 4
Thread[140441891612416] has found 22044 prime numbers in (1 ~ 250000).
Thread[140441883219712] has found 19494 prime numbers in (250001 ~ 500000).
Thread[140441806698240] has found 18700 prime numbers in (500001 ~ 750000).
Thread[140441798305536] has found 18260 prime numbers in (750001 ~ 1000000).
The main thread[140441891616576] has found 78498 prime numbers in (1 ~ 1000000).
yeonggi@DESKTOP-QK10V32:~/system/hw4$
```