

시스템 프로그래밍

argc, argv

소프트웨어학과 2022764034 이이슬

argc, argv 들어가기 전

명령행 인자(Command Line Arguments)

```
#include <stdio.h>
```

```
int main(void){  
    return 0;  
}
```

```
#include <stdio.h>
```

```
int main(int argc, char* argv[]){  
    return 0;  
}
```

프로그램을 실행할 때 명령 프롬프트나 터미널을 통해 프로그램에 전달되는 인자 값

이 인자들은 프로그램이 실행될 때 프로그램 이름 뒤에 공백으로 구분하여 입력

argc, argv 들어가기 전

argument(전달인자) vs parameter(매개변수)

argument(전달인자)

- 함수나 메서드를 호출할 때 실제로 전달되는 값

parameter(매개변수)

- 함수나 메서드 정의에 사용되는 변수
- 이 변수들은 함수가 실행될 때 필요한 입력 값의 '형식'을 정의

Ex)

- `int add(int a, int b) ->` a와 b는 parameter
- `add(3,5) ->` 3과 5는 argument



argc, argv

```
int main(int argc, char* argv){  
    return 0;  
}
```

argc (Argument Count)

- 이 변수는 프로그램에 전달된 인자의 개수를 나타냄
- 프로그램 이름 자체도 하나의 인자로 취급되기 때문에 값이 항상 1 이상

argv (Argument Vector)

- 이 배열은 문자열 포인터의 배열로 실제로 전달된 인자들의 값을 담음
- argv[0]은 프로그램의 실행 경로를 나타내며,
argv[1]부터 시작하는 나머지 요소들은 사용자가 입력한 인자들을 순서대로 담음



```
iseul@DESKTOP-Q9K644I:~/ex1$ vi test.c
iseul@DESKTOP-Q9K644I:~/ex1$ cat test.c
#include<stdio.h>

int main(int argc, char* argv[]){
    printf("인자 수 : %d\n",argc);

    for(int i=0;i<argc;i++){
        printf("argv[%d] = %s\n",i,argv[i]);
    }

}
iseul@DESKTOP-Q9K644I:~/ex1$ make test
cc      test.c      -o test
iseul@DESKTOP-Q9K644I:~/ex1$ ./test 시스템프로그래밍 화요일 5시부터 8시까지
인자 수 : 5
argv[0] = ./test
argv[1] = 시스템프로그래밍
argv[2] = 화요일
argv[3] = 5시부터
argv[4] = 8시까지
```



참고자료

<https://wonlf.tistory.com/entry/argument%EC%99%80-parameter%EC%9D%98-%EC%B0%A8%EC%9D%B4%EC%A0%90-%EA%B7%B8%EB%A6%AC%EA%B3%A0-argc-argv-main%ED%95%A8%EC%88%98%EC%9D%98-%ED%8C%8C%EB%9D%BC%EB%AF%B8%ED%84%B0%EC%9D%98-%EB%8C%80%ED%95%98%EC%97%AC>
<https://www.acmicpc.net/board/view/7154>
<https://blog.naver.com/ygszzang11/50178777730>
<https://1y9u9j2in.tistory.com/163>
<https://m.blog.naver.com/skssim/119044035>
<https://velog.io/@sjmh0507/CS-%EB%AA%85%EB%A0%B9%ED%96%89-%EC%9D%B8%EC%9E%90>
<https://www.youtube.com/watch?v=JBSA6DZNyec>

감사합니다 :D