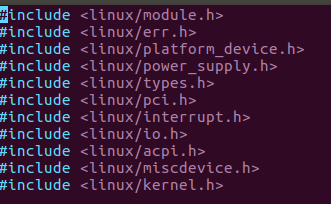
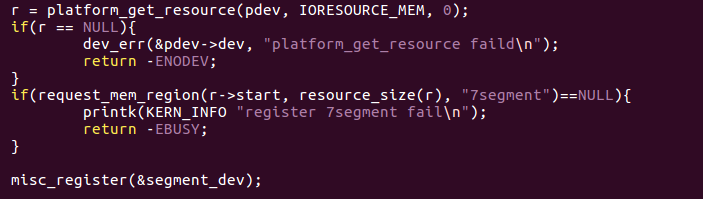
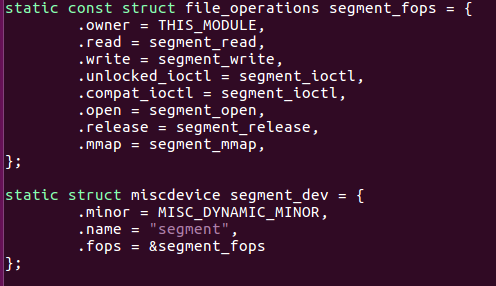
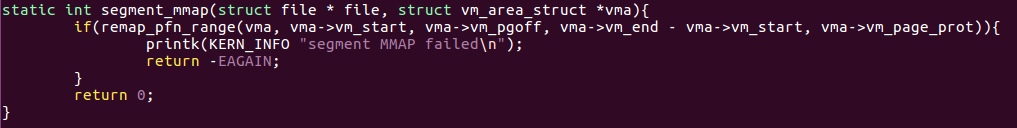
Embedded System Practice Lab 3

2016311821 한승하

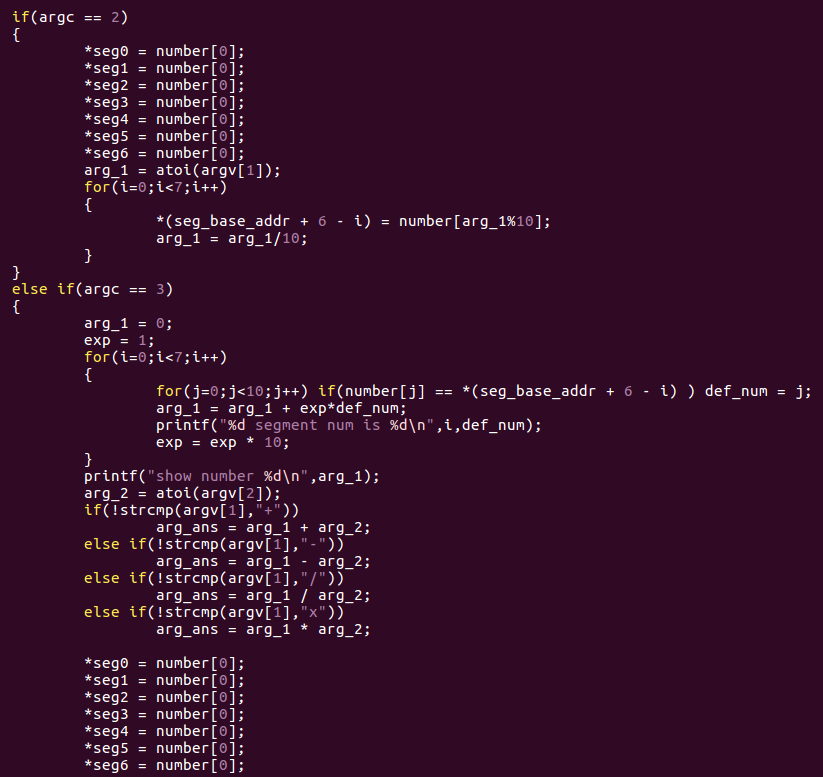
우선 이번주 Exercise를 진행하기 위해 실습 수업에서 주어진 대로 환경 설정을 완료하였습니다.

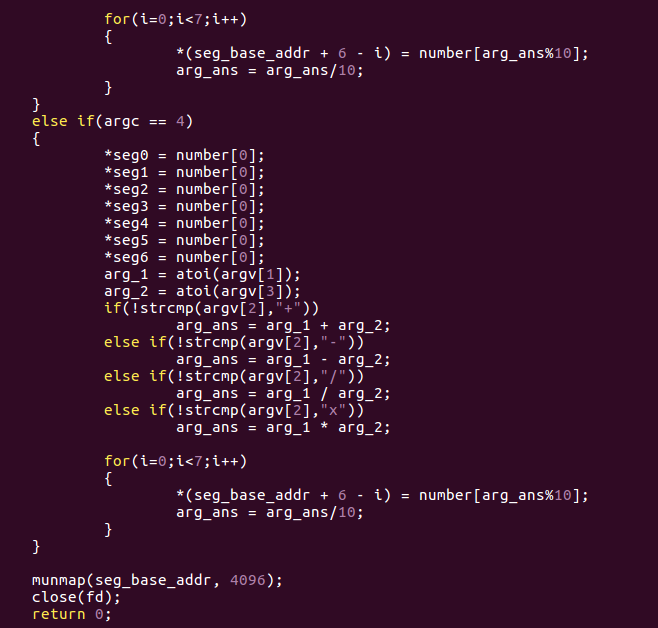
먼저 mmap을 이용한 mini calculator입니다.

<Using Mmap>

 Mmap함수를 사용하기 위한 코드들을 goldfish\_segment.c파일에 추가해 주었습니다.

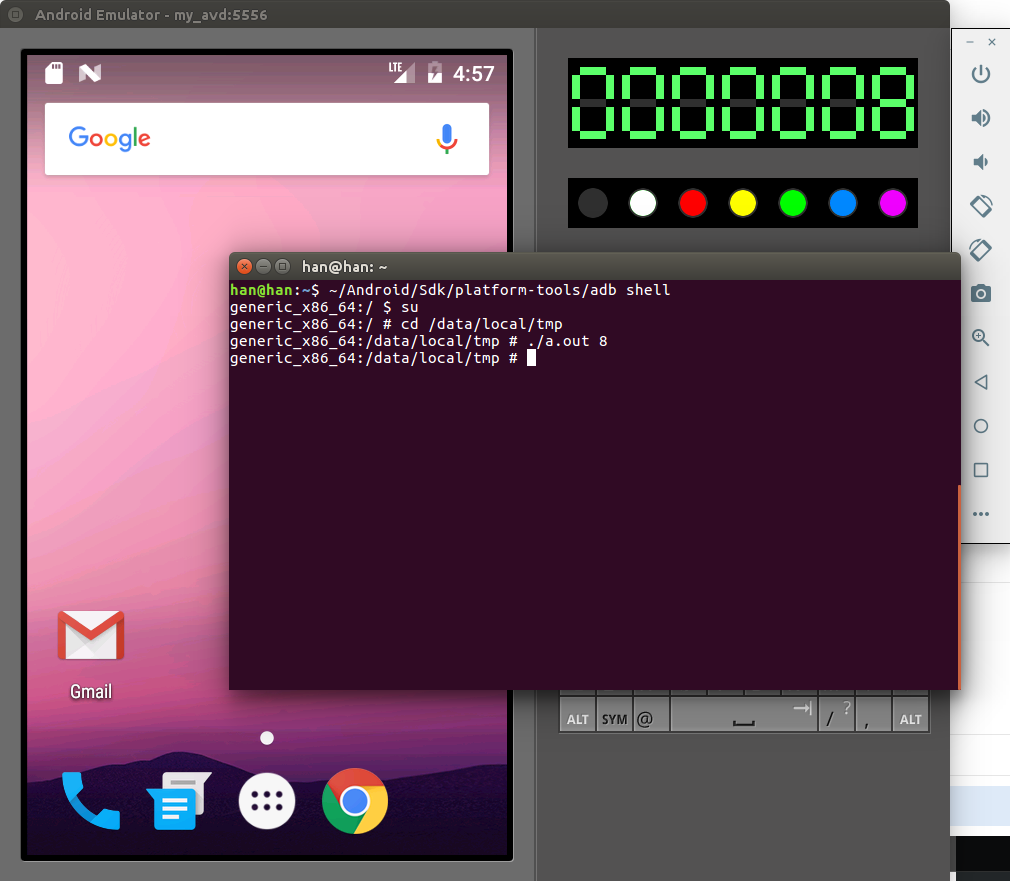
<User Code>

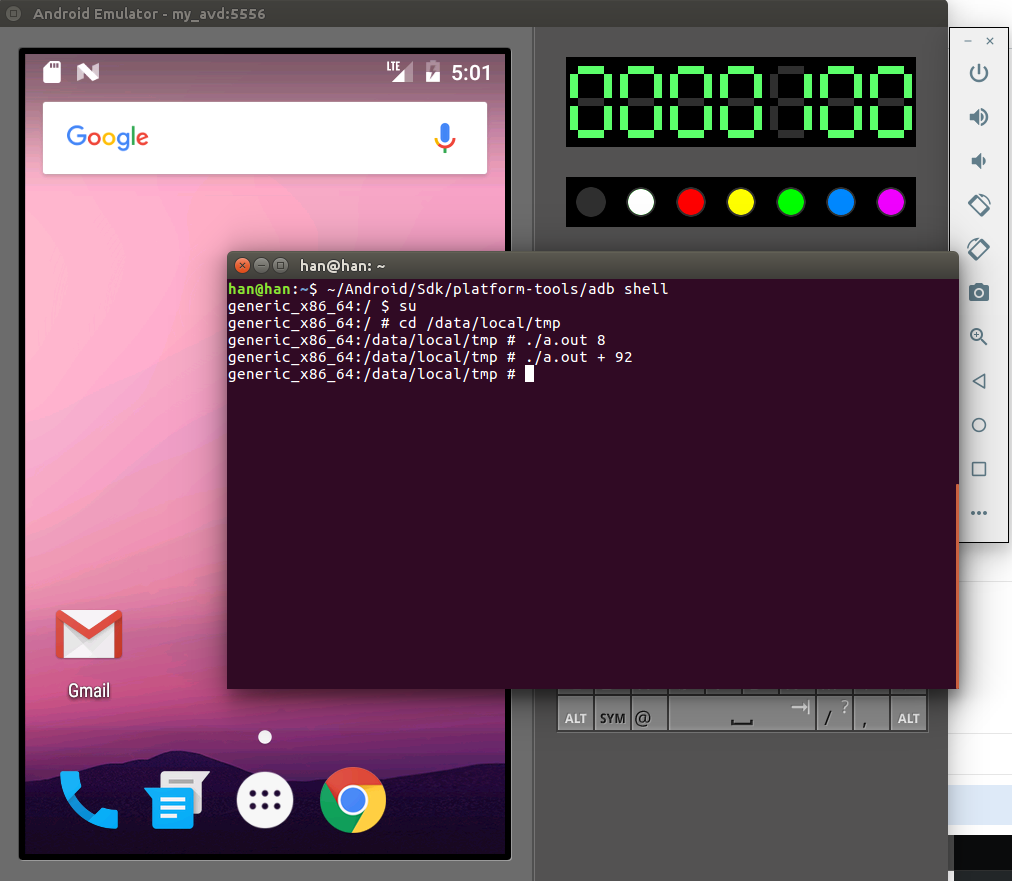


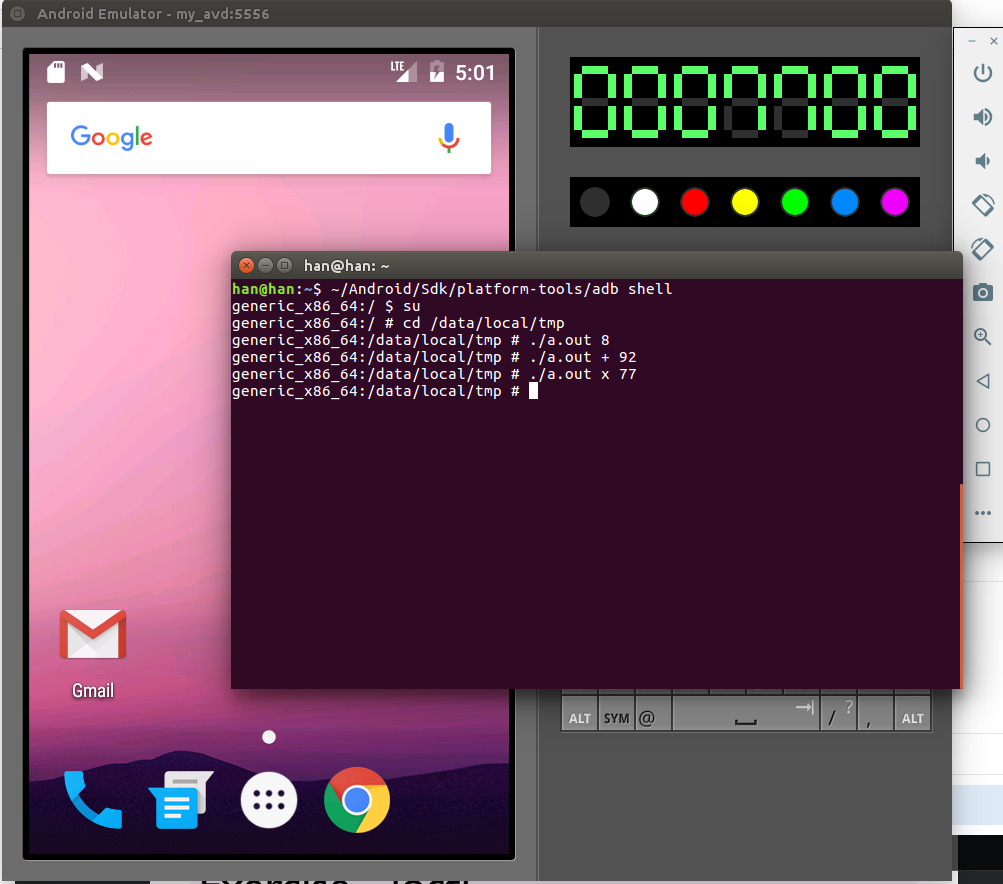


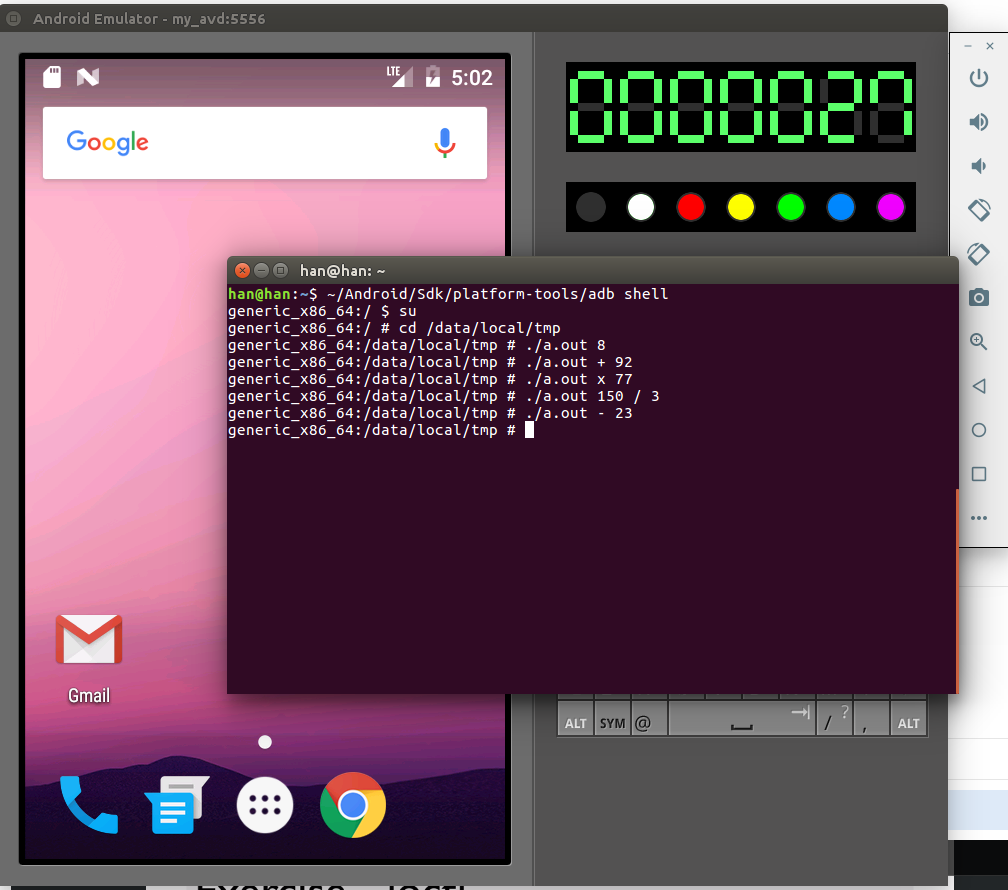
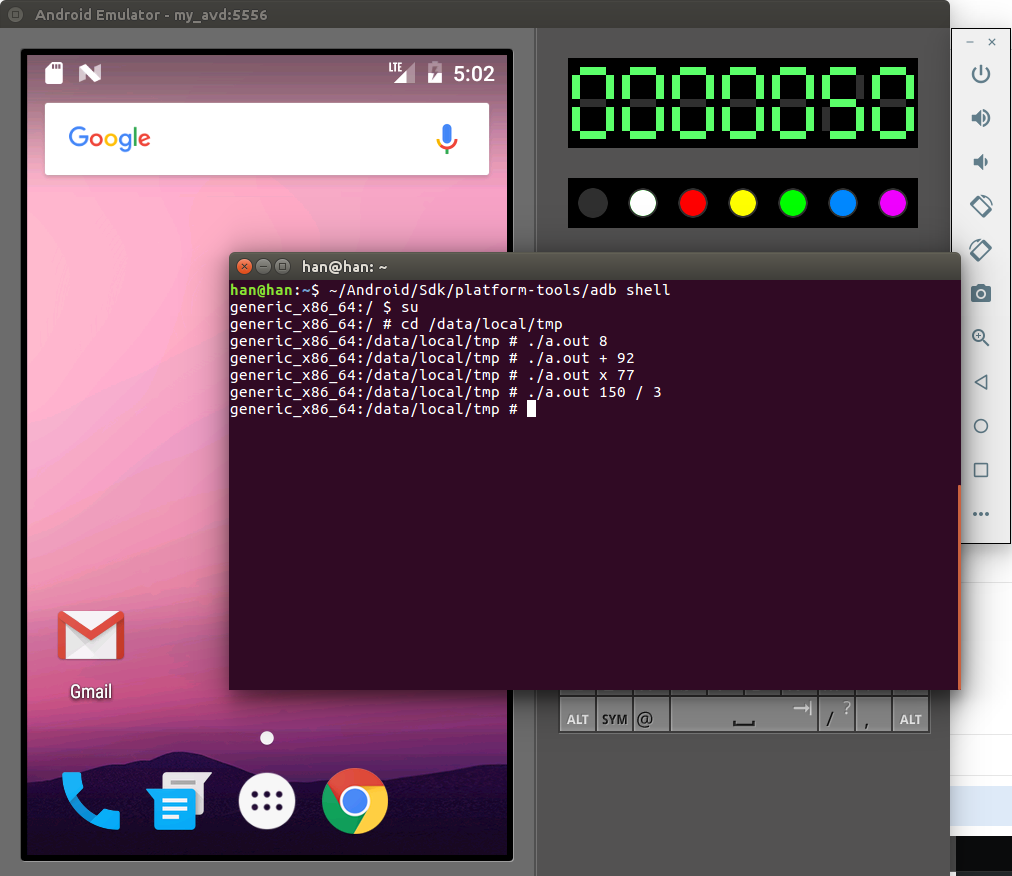
mmap함수를 이용한 mini\_calculator usercode입니다. 각 argv상황들에 맞게 처리해 주었으며, mmap된 메모리 영역을 \*seg\_base\_addr를 base로 하여 각 segment에 필요한 숫자들이 출력될 수 있게 하였습니다.

<Testing>



위는 8을 입력한 상황입니다. 

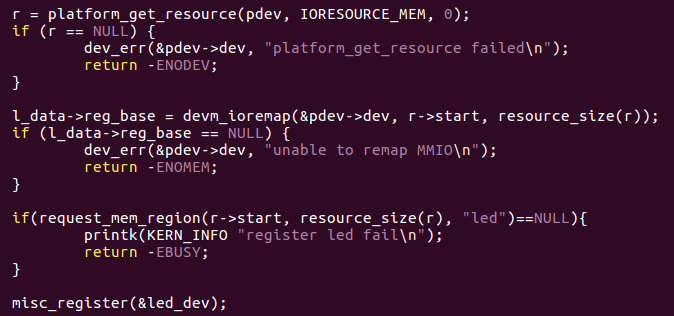
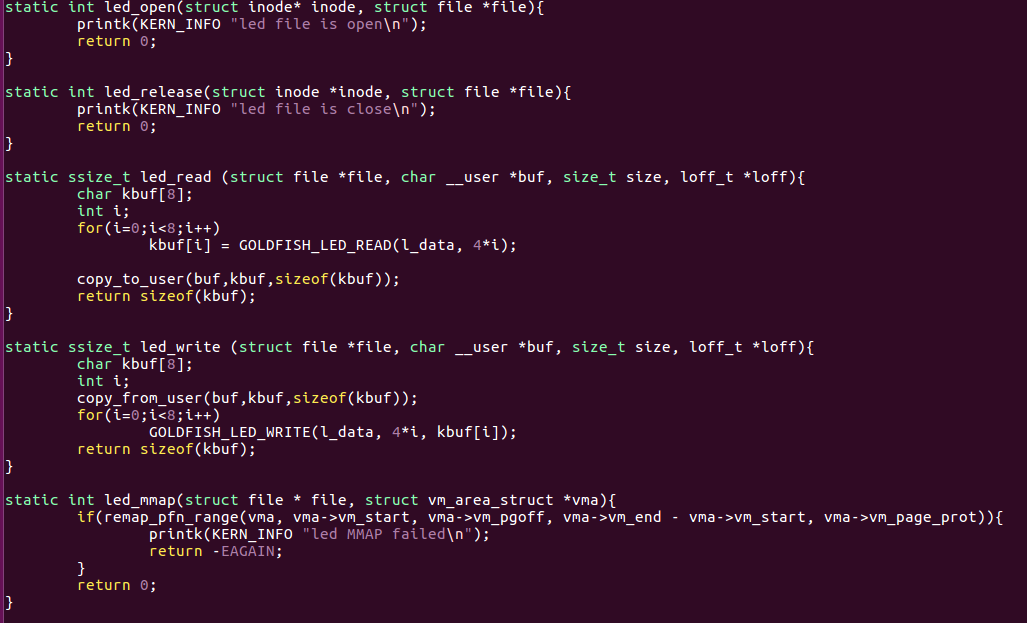
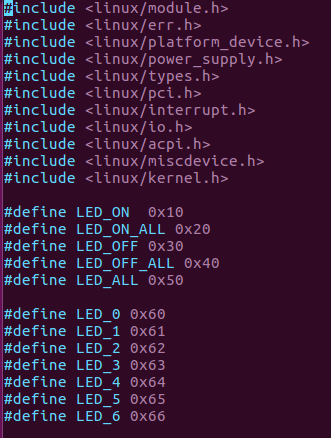
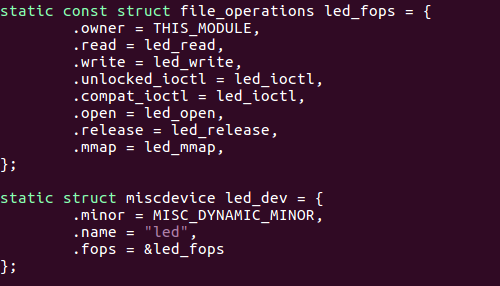


그 이후 차례대로 + 92, x 77을 입력한 상황의 출력입니다.

마지막으로 150 / 3과 – 23을 입력하여 출력을 보는 것으로 mmap calculator에 대한 testing을 마무리 하였습니다.

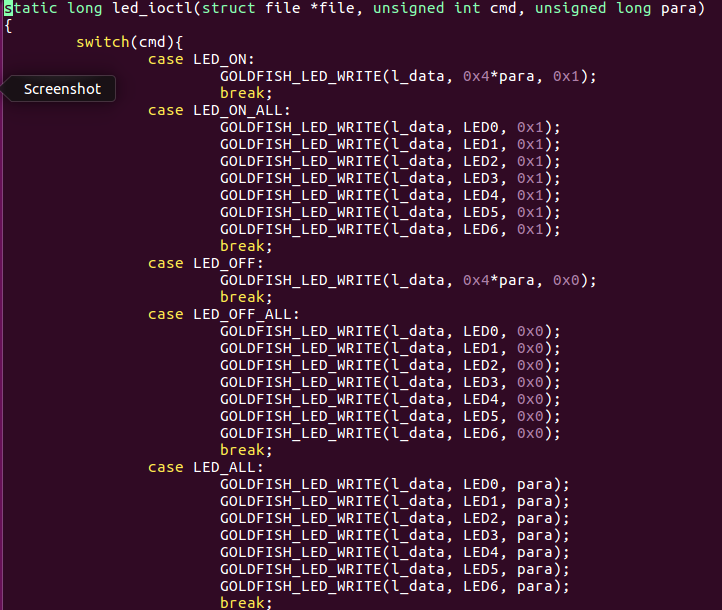
다음은 ioctl을 사용한 LED출력입니다.

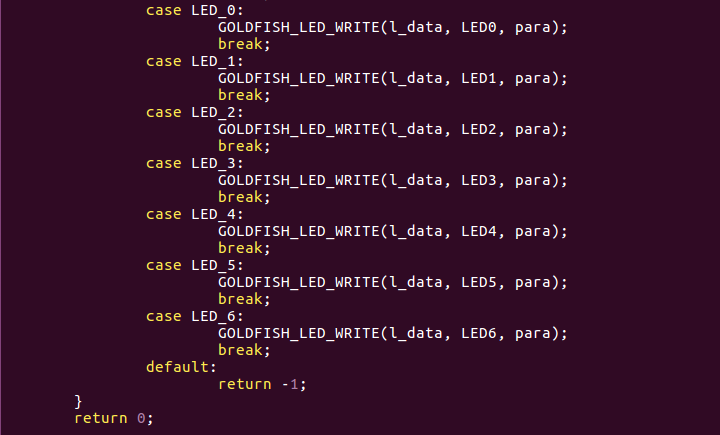
<Using Ioctl>



위의 코드들을 goldfish\_led.c에 추가해줌으로써 ioctl을 사용할 준비를 하였습니다.

<Ioctl Code>

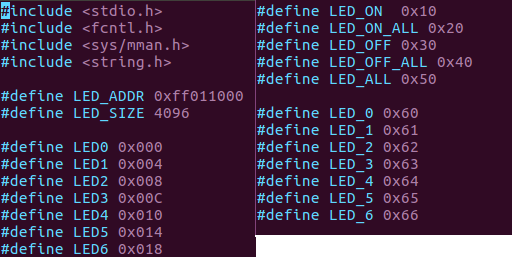


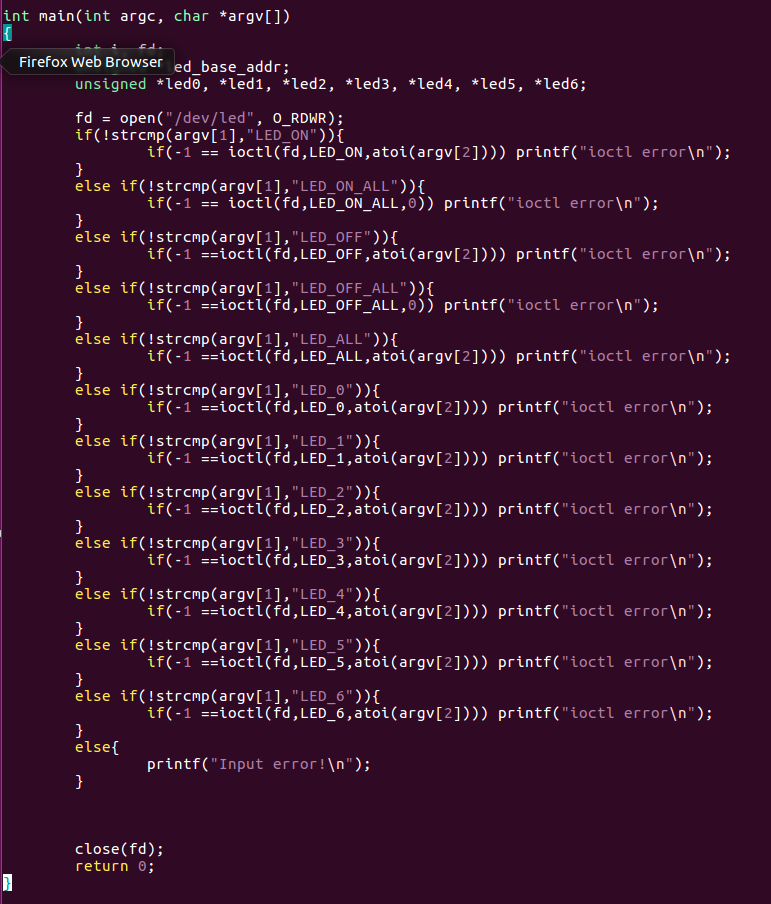


위 코드는 goldfish\_led.c에 작성한 led\_ioctl code입니다. 각 command ( LED\_ON, LED\_ON\_ALL, LED\_OFF, LED\_ALL, LED\_0~6 )에 대해 Case를 나눠 동작하게 하였으며, LED\_ON, LED\_OFF에선 LED번호를, LED\_ALL, LED\_0~6에선 색의 번호를 Parameter로 받아 사용하였습니다.

일반적인 상황에선 0을 Error인 예외상황에선 -1을 return 합니다.

<User Code>



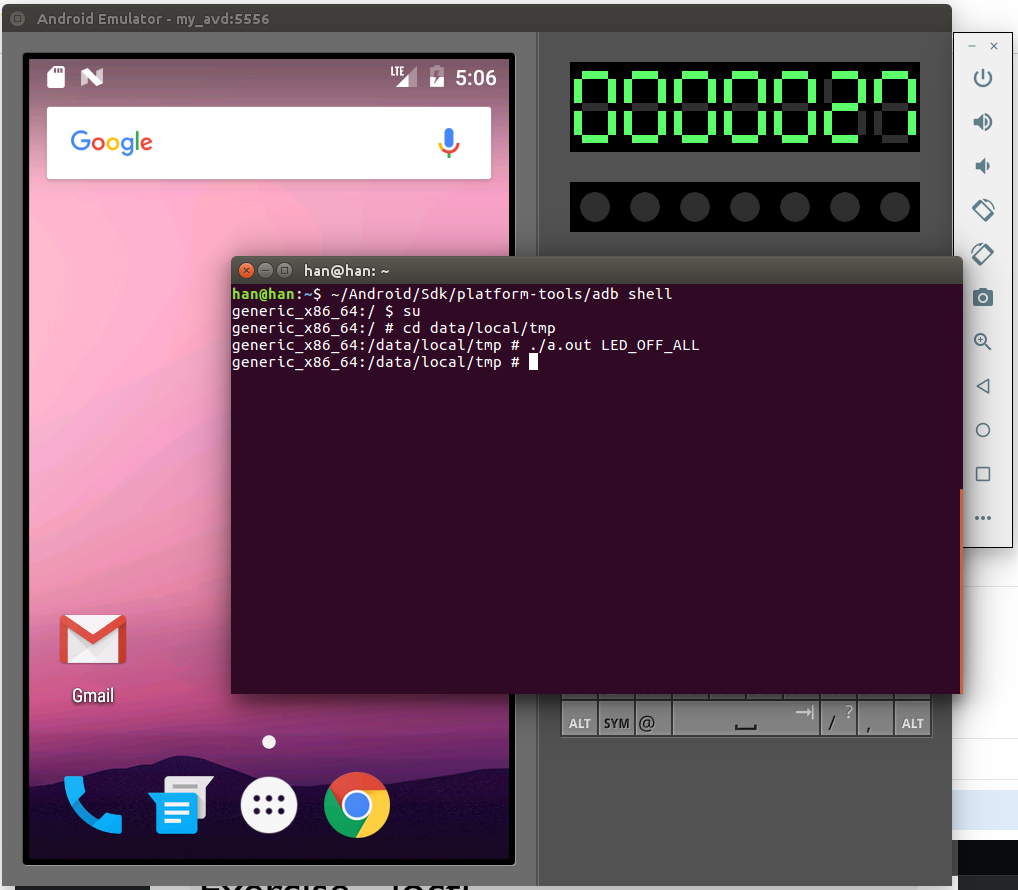


위는 LED를 위한 Usercode입니다.

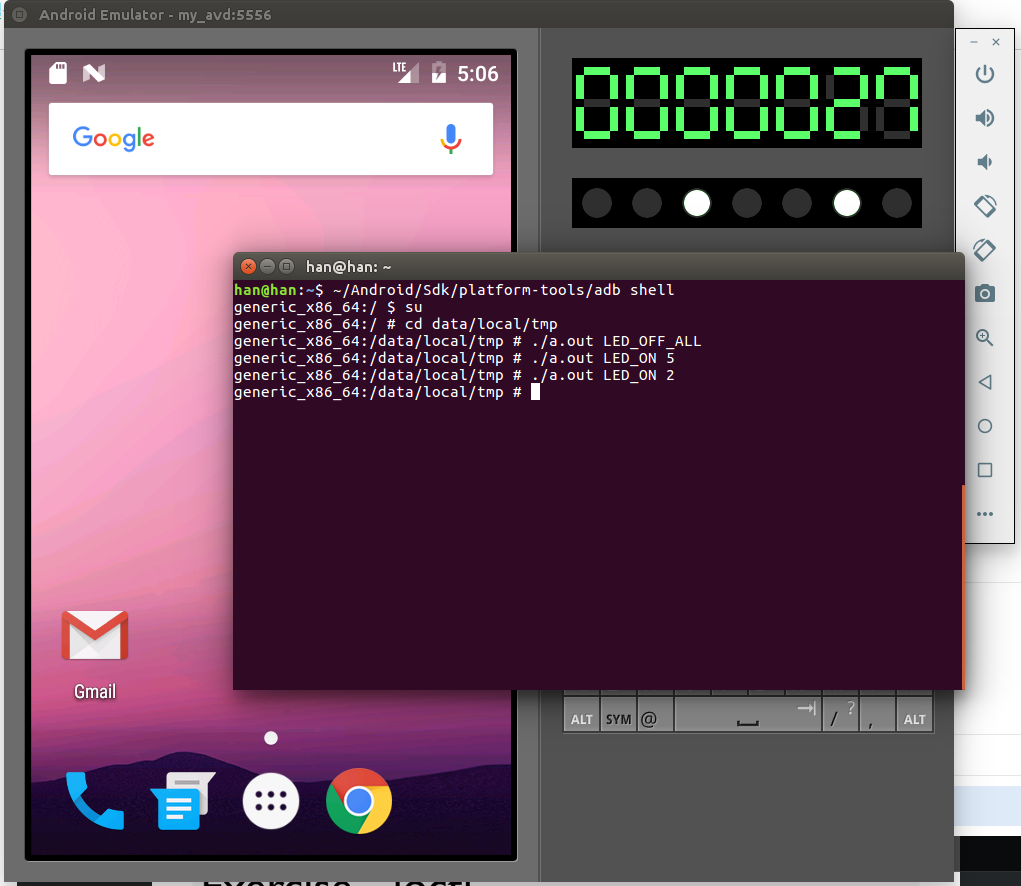
Ioctl과 맞게 Command 별로 분류하여 작성하였으며, 넘겨줘야 하는 parameter들은 입력을 추가로 할 수 있게 작성하였습니다. Ioctl의 return값에 따라 error을 출력할 수 있게 작성하였습니다.

LED OFF는 gray, LED ON은 white색을 사용하였습니다.

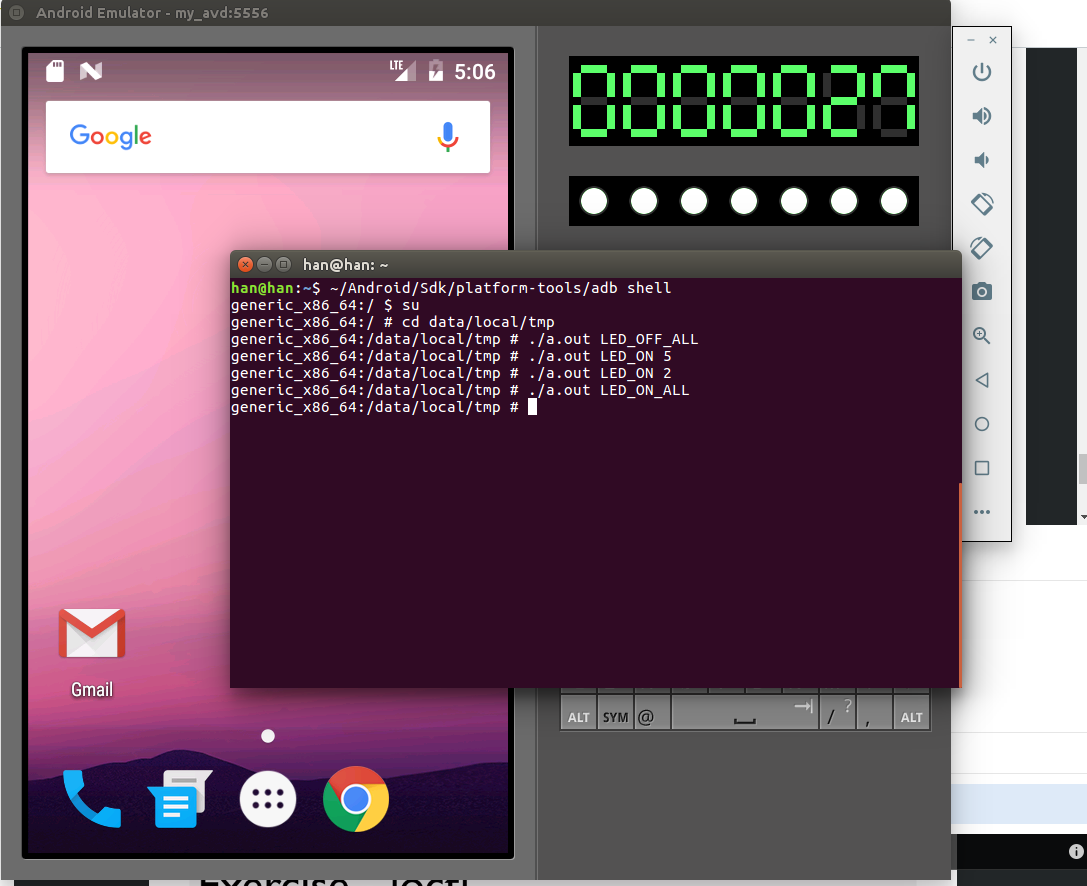
<Testing>



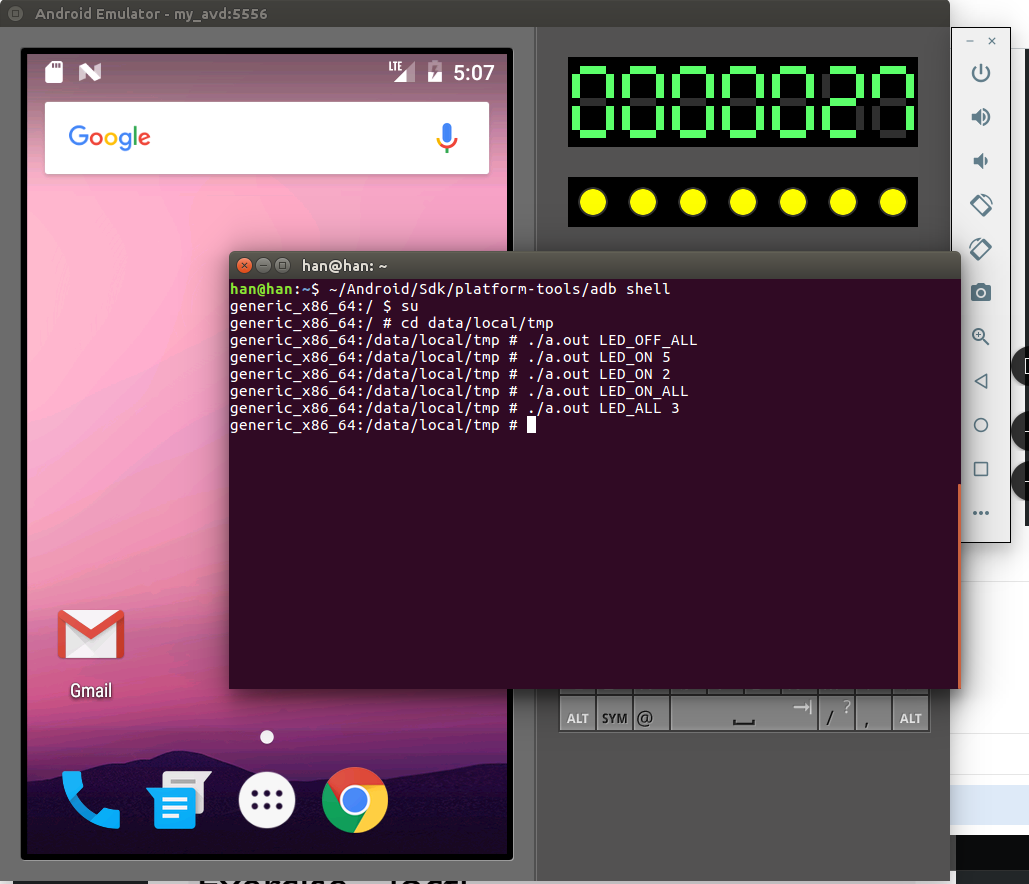
LED\_OFF\_ALL Command입니다.



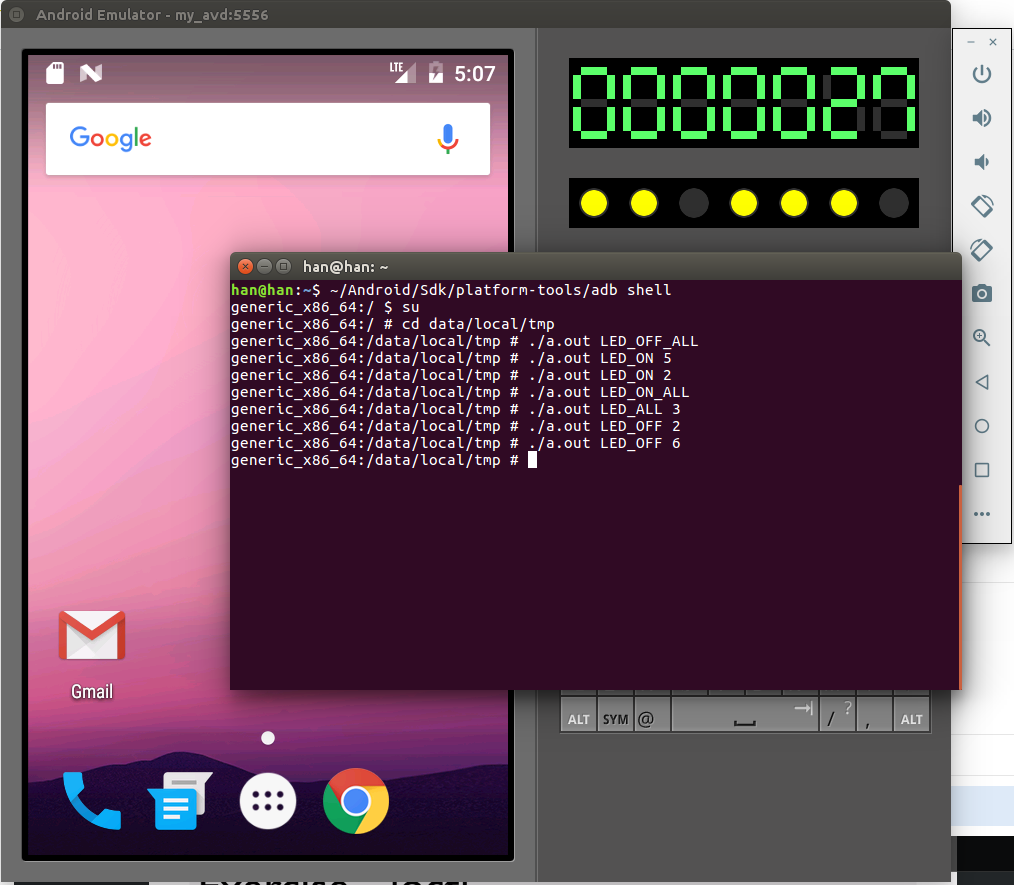
LED\_ON 5, LED\_ON 2로 5, 2번 LED를 점등한 상황입니다.



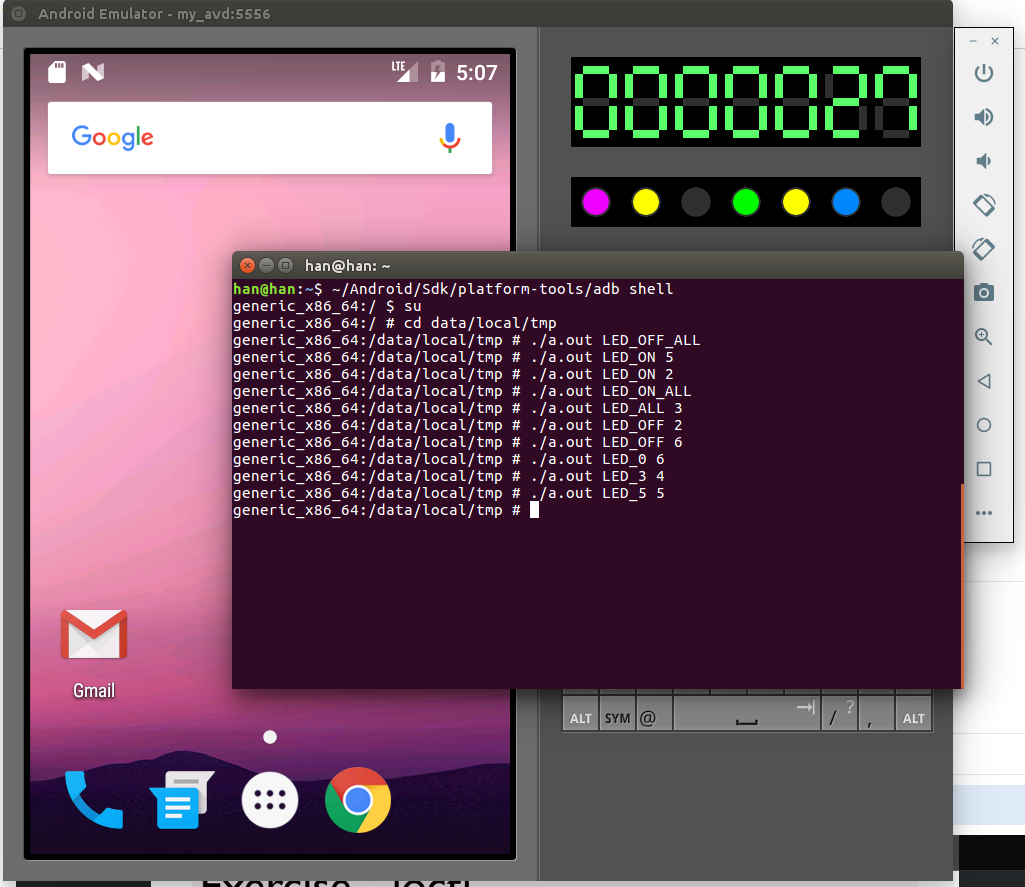
LED\_ON\_ALL으로 모든 LED를 점등한 상황입니다.



LED\_ALL 3으로 모든 LED를 3번색인 yellow로 바꿔준 상황입니다.



LED\_OFF 2, LED\_OFF 6을 통해 2번, 6번 LED를 꺼준 상황입니다.



마지막으로 LED\_0 6, LED\_3 4, LED\_5 5로 0번을 purple, 3번을 green, 5번을 sky색으로 바꿔준 모양입니다.

감사합니다.