전공 : 컴퓨터공학과

학년 : 2학년

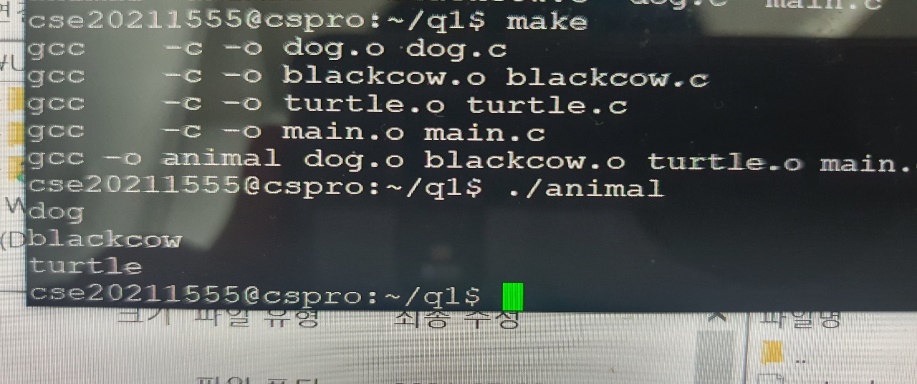
학번 : 20211555

이름 : 유시온

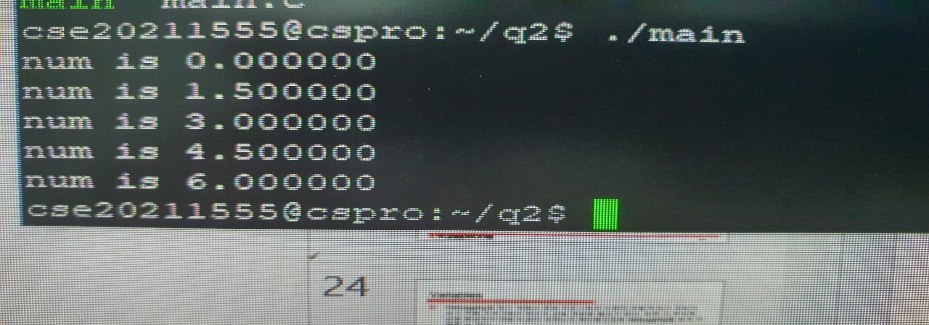
1. 목적

-1. 실습 결과화면을 첨부한다.

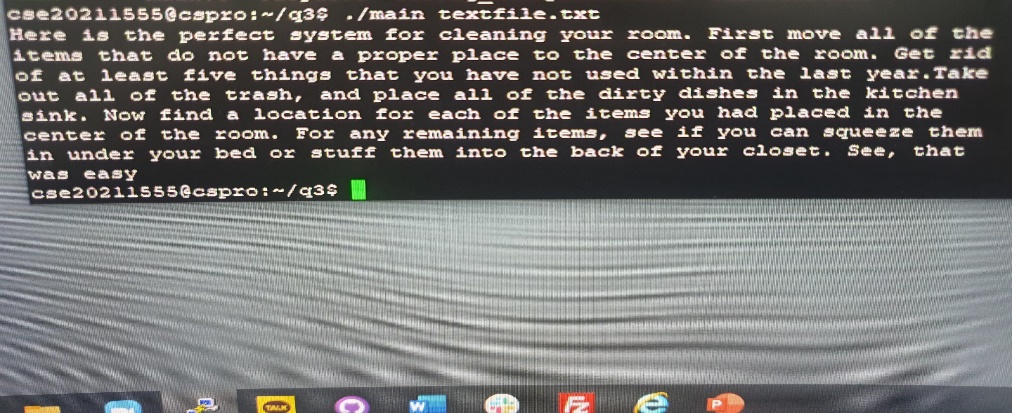
1번



2번



3번



-2. Fmt를 구현하기 위해 사용한 함수들과 그 함수들의 목적을 간단히 설명한다.

Remove\_Blanks\_At\_The\_End( char \*line ) : main함수에서 입력받은 line에 대하여 문장 끝이 \n으로 끝나면 문자열의 끝 공백 바로 이전의 문자로 간 후 그 다음 자리를 줄바꿈으로 해준 뒤 공백을 채워준다. 만약 문장 끝이 ‘\0’으로 끝났다면 문자열의 끝 공백 바로 이전의 문자로 간 뒤 그 다음자리를 ‘\0’으로 바꿔준다

Print\_Line( char \*line, int \*Count, int \*B\_Flag ) : 출력 역할의 함수; 문자열을 길이에 맞게(N\_Blank,N\_Chars를 통해 계산)끔 출력

Get\_Blanks\_Chars( char \*line, int Start, int \*N\_Blanks, int \*N\_Chars ) : 주어진 line을 ‘\n’또는 ‘\0’가 나올때까지 검사해서 space가 아닐 땐 \*N\_Chars를 1씩 더해주며 blank\_flag를 1로 설정해주고 만약 space가 나왔는데 이전까지 아무글자도 없었을 경우엔 Print\_Line에서 오류를 표시하게끔 하고 만약 이전에 글자가 있었을 경우엔 그냥 넘어간다.

-3. 실습시간에 작성한 Makefile의 한줄 한줄 의미를 설명한다.

CC=gcc

(gcc를 대신하여 CC로 쓰도록 매크로 정의(CC는 c 컴파일러에 대한 매크로로 미리 정해져 있음))

TARGET = animal

(animal파일을 TARGET으로 매크로 정의)

OBJECTS = dog.o blackcow.o turtle.o main.o

(dog.o blackcow.o turtle.o main.o 파일들을 한번에 OBJECTS로 매크로 정의)

$(TARGET) : $(OBJECTS)

(목표(target)과 의존관계를 설정해준다 여기선 animal이 target파일이 되고 의존관계 부분에 정의된 파일들은 dog.o blackcow.o turtle.o main.o이다)

$(CC) -o $(TARGET) $(OBJECTS)

(gcc -o animal dog.o blackcow.o turtle.o main.o 매크로를 통해 정의한 내용 그대로 치환 해서 사용 가능하다)

$(OBJECTS) : animal.h

(헤더파일 할당해주기 파일이 변경되어도 object파일을 새로 생성해줌)

.PHONY : clean

(.PHONY의 명령어를 정의 여기서는 clean으로 정의하였다)

clean :

rm $(TARGET) $(OBJECTS)

(make clean을 썼을 때 rm(삭제) $(TARGET) $(OBJECTS) (이 매크로에 해당하는 것들)을 의미 )

-4. 규칙 R5를 어떤 알고리즘으로 구현하였는지 상세히 설명한다.

Remove\_Blanks\_At\_The\_End에서 blank를 검사하고 그에 따라 k-1을 마지막 글자로 설정하여 그 다음을 공백으로 할지 줄바꿈 후 새 줄에 출력하고 첫 부분의 blank를 채워줄 지를 결정한다.

-5. Make의 옵션들에 대하여 정리한다.

-d : debug로써 make하면서 정보를 출력한다

-k : 에러가 발견되어도 실행한다

-h : make가 지원하는 옵션들 보여주기

-n : 실행할 명령 보여주기 그러나 실행하지는 않는다.

-q : 명령과 출력 없이 종료 상태값만 반환하도록한다.

-v : make 버전, 저작권 등등 상세정보 출력

-l dir : makefile을 찾기 위한 디렉토리 지정