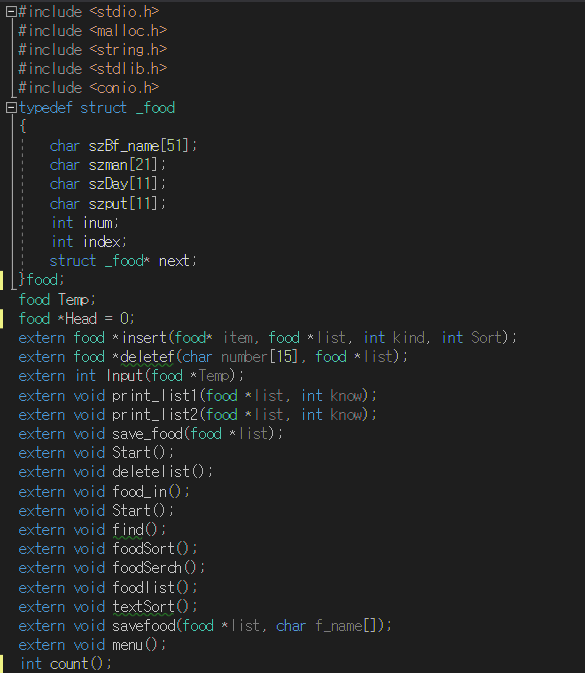
임베디드SW기초

식품보관시스템

201801730 김다희

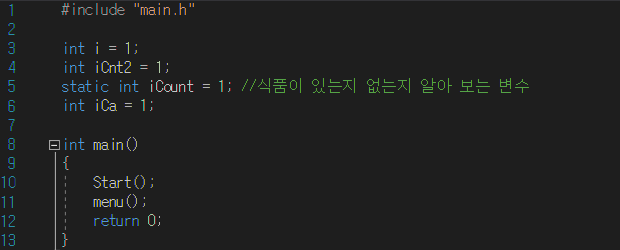
<코드>



<헤더 파일>

구조체를 이용하여 식품의 이름, 주인명, 저장날짜, 유통기한, 순번, 개수를 저장할 수 있도록 만들고, SLL를 이용하여 저장한 식품 노드들을 연결할 수 있도록 만든다.

각 함수들을 선언해준다. 함수는 추가, 저장, 삭제선택, 삭제, 시작, 메뉴, 정렬, 검색 등등 총 18개이다. 각 함수들의 설명은 아래와 같다.



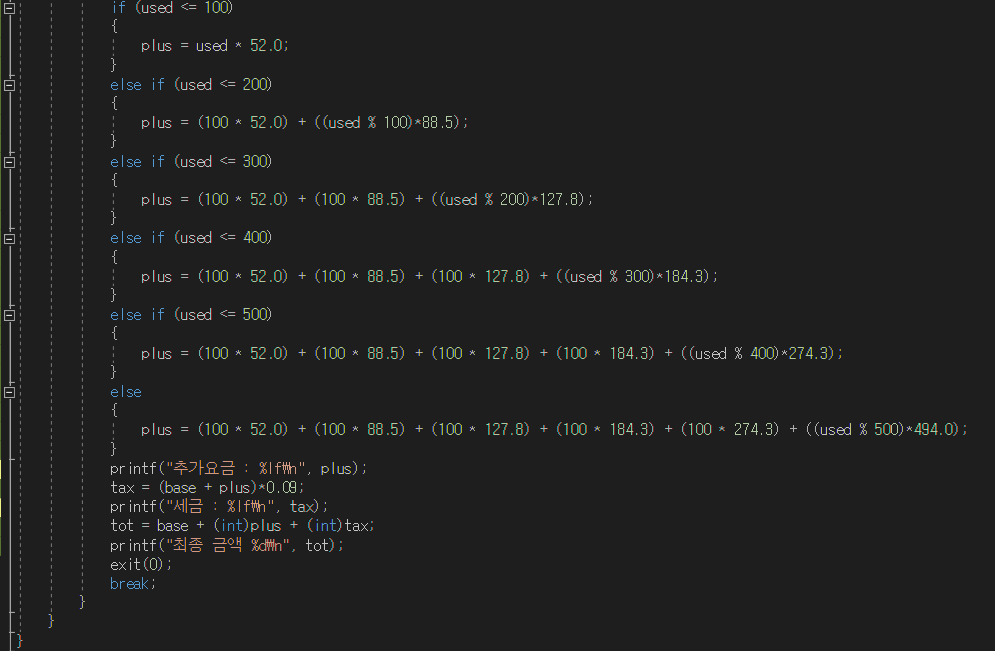
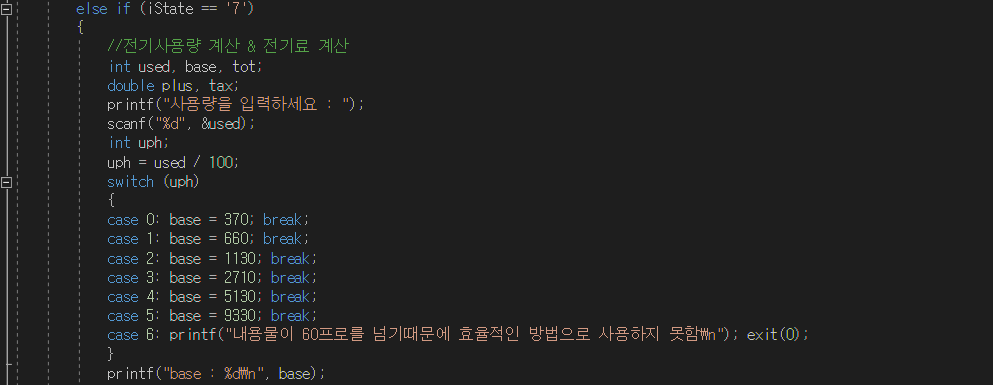
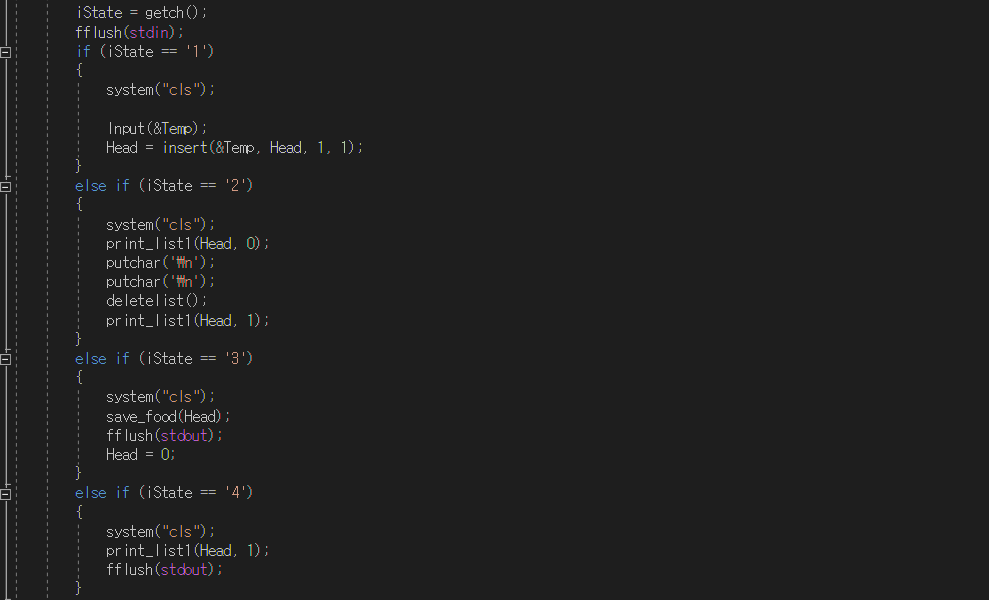
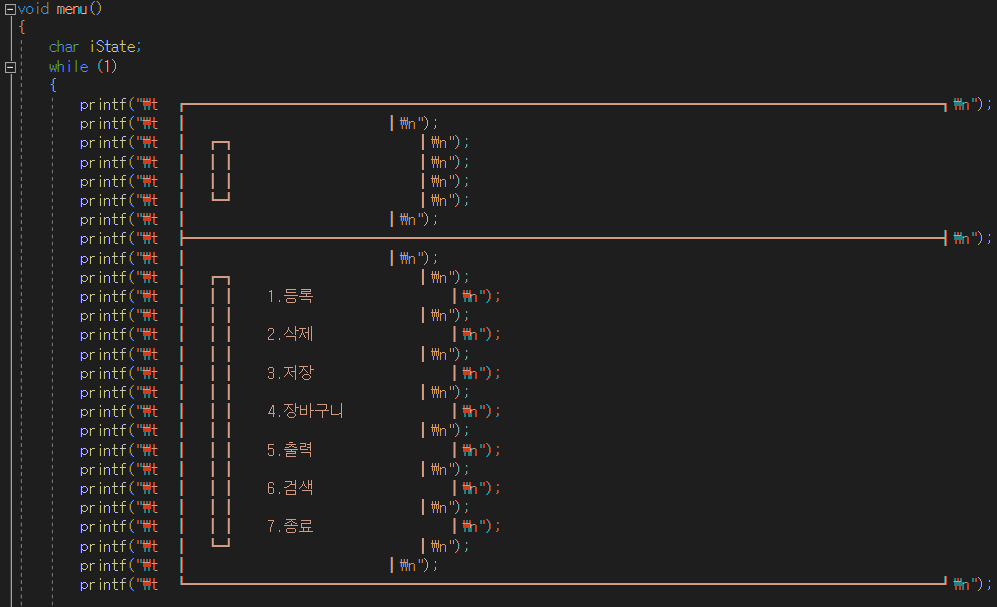
<전역변수 및 main 함수>

식품을 저장하거나 정렬을 할 때 순번을 알려줄 변수와 식품이 있는지 없는지를 알려줄 변수를 를 전역변수로 선언한다. main함수는 처음에 시작화면을 출력해주고 start함수에서 강제로 종료하지 않는다면 menu함수로 넘어갈 수 있도록 설정한다.



<start 함수>

시작화면을 출력하고 만약 q를 누른다면 시스템이 종료하고 다른 키를 누른다면 시스템을 cls하여 menu화면으로 넘어가도록 만든다.

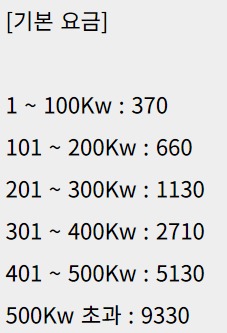


<menu 함수>

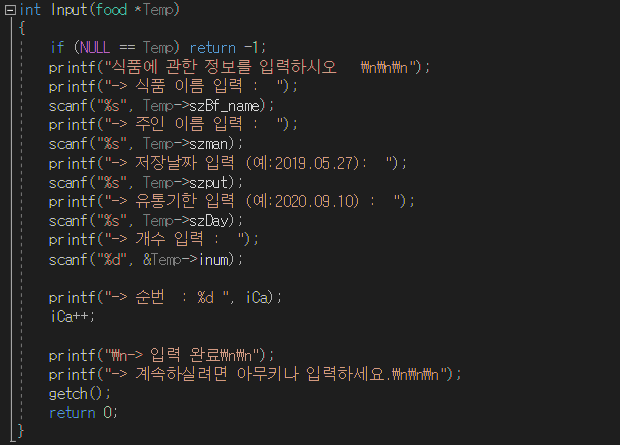
Menu는 등록, 삭제, 저장, 장바구니, 출력, 검색, 종료로 통 7가지의 기능을 넣는다. 1부터 6까지의 기능들은 각 기능을 하는 함수를 이용한다. 예를 들어 등록인 1번을 입력으로 준다면 input함수를 불러들여 식품의 정보를 입력하고 insert함수를 이용하여 입력한 정보를 memory를 할당한 다음 식품들의 data가 들어있는 곳에 삽입한다. 나머지는 이와 같이 작동한다. 더 상세한 함수들의 설명은 아래에 있다.

7번은 시스템을 종료하는 것으로 종료를 하면 전기사용량과 전기료를 계산하여 출력을 해준다. 자신이 사용한 양을 받아들인 후, 그 사용량에 따른 base요금을 알려주고, 사용량에 따른 외부요인으로 인한 추가요금을 등차와 등비를 이용하여 구해준 뒤, base와 추가요금에 따른 세금을 더해 최종 전기료를 계산해준다.

(전기요금 : 기본 요금+사용량\*KW당 요금+세금)

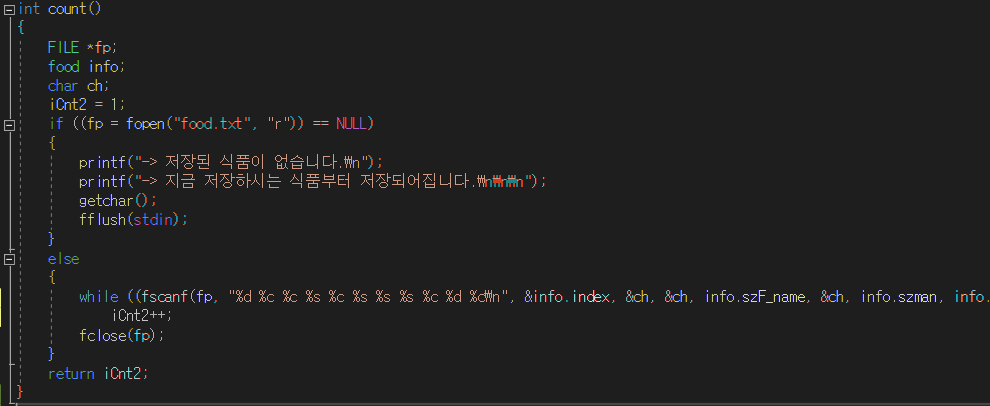
5

위에 표처럼 기준을 나누어 기본 요금과 추가 요금을 계산했다.



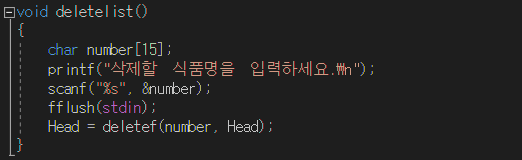
<input 함수>

menu에서 1번을 입력하여 등록을 할 때 사용하는 함수로, 구조체를 사용하기 때문에 Temp->쓴 후 저장할 식품명, 주인 이름, 저장날짜, 유통기한, 개수를 입력 받는다.



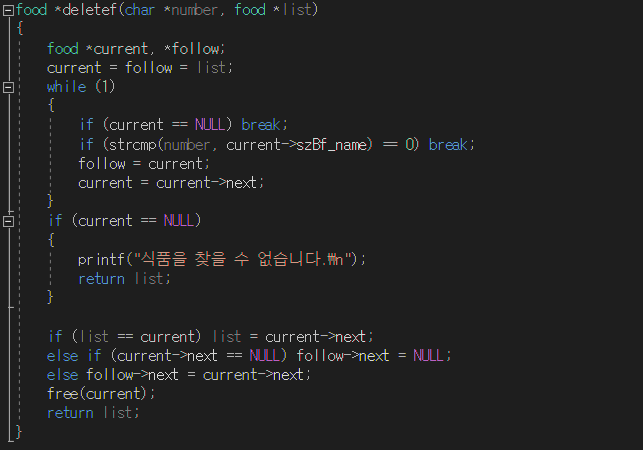
<count 함수>

식품이 저장될 때 불려지는 함수로, 식품의 순번을 리턴해준다. Txt에 저장된 식품이 없다면getchar를 사용하여 표준 입력에서 다음 문자를 리턴해준다. 저장된 식품이 있다면 반복문을 txt이 EOF 즉, 파일 끝에 도달할 때까지 cnt를 돌려 몇번째인지 계산하여 값을 리턴해준다.



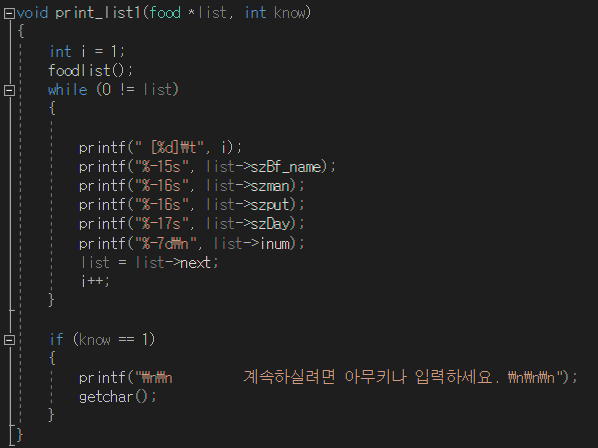
<deletlist 함수>

menu에서 2번을 입력하여 삭제할 때 사용하는 함수로, 식품명을 입력한 후 SLL를 이용하여 등록된 데이터를 삭제해주는 deletef함수를 불러온다.



<deletef 함수>

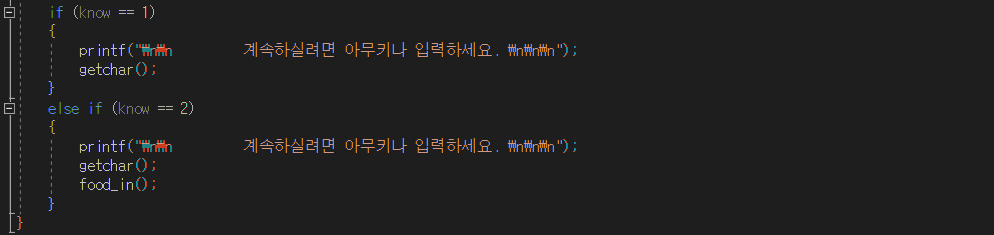
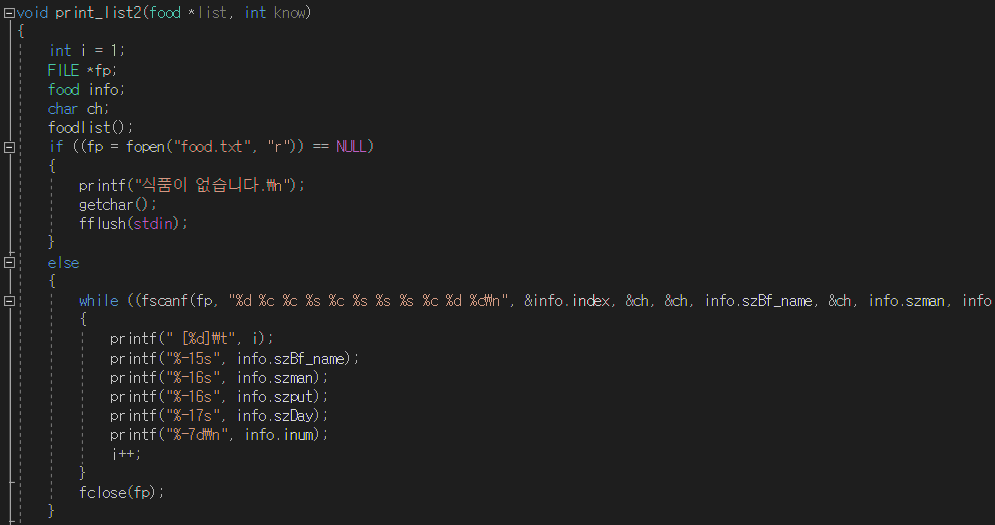
SLL를 이용하여 등록된 식품의 데이터를 삭제해주는 함수로, node를 하나씩 이동시켜 strcmp를 이용하여 입력한 식품과 동일한 식품이 있는지 검사한다. 있다면 해당 node를 건너뛰어 연결한 다음 free를 이용하여 저장된 식품의 정보를 메모리할당을 해체하여 삭제하고, 없다면 식품을 찾지 못했다고 출력해준 다음 기존의 list를 리턴해준다.



<printf1 함수>

menu에서 4번을 입력했을시 장바구니의 등록된 식품들의 정보를 알려주는 함수이다.

(4번의 장바구니는 식품 배송서비스로 냉장고의 식품 저장과는 다른 의미이다. 따라서 등록한 후 저장을 안할시 냉장고의 저장된 것이 아니기 때문에 식품 배송 서비스라고 인식을 하기 때문에 출력을 누르면 등록된 것은 보이지 않고 장바구니를 눌러야 등록한 것을 볼 수 있다.)



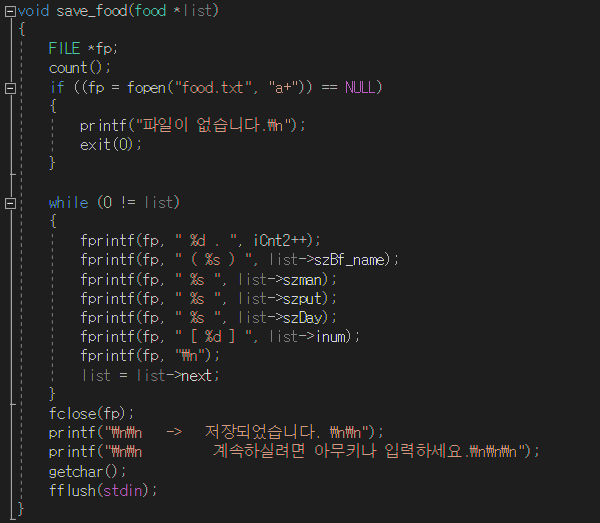
<printf2 함수>

menu에서 5번을 입력시, 냉장고에 저장되있는 식품들의 정보를 출력해주는 함수이다. 또한 menu에서 6번 입력시, 원하는 식품의 정보를 검색해줄 때도 사용하는 함수이기도 하다. Know가 1일 경우는 단순 출력, 2일 경우는 검색과 출력의 메뉴로 가게 해준다.



<save 함수>

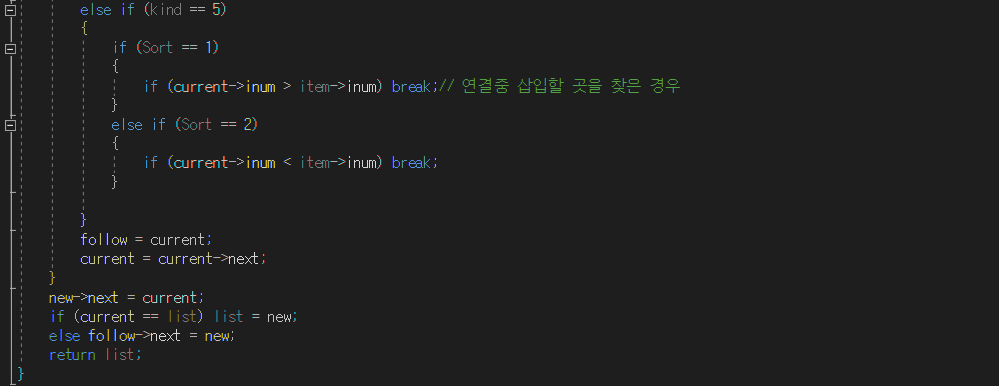
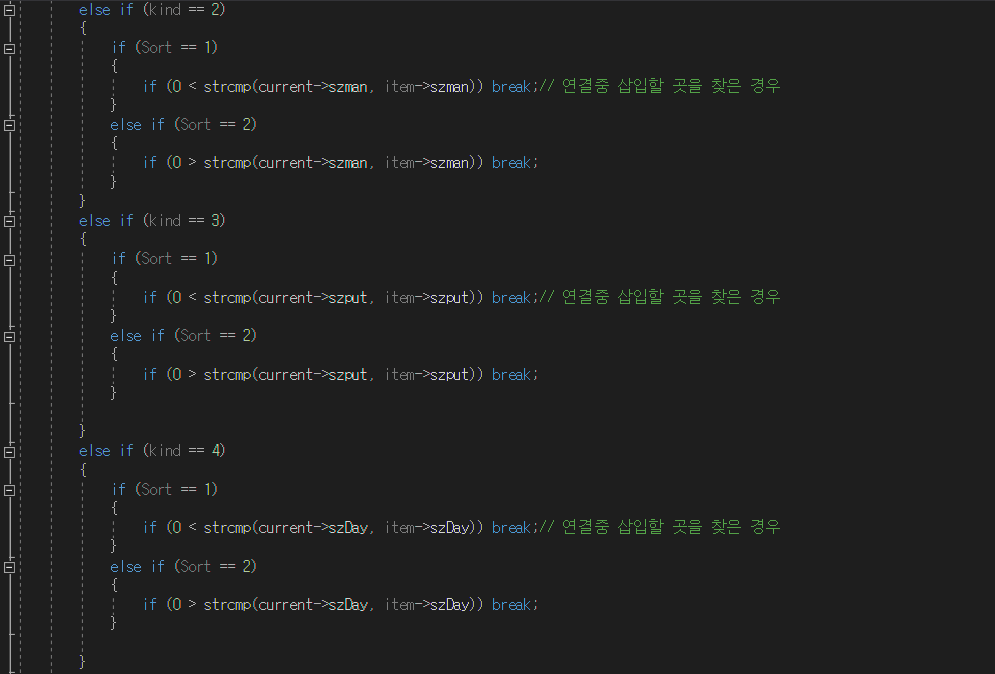
식품의 정보를 수정하거나 검색하거나 정렬을 하여 자신이 원하는 식품의 data만 뽑아서 저장할 때 사용하는 함수이다. 새로운 data list를 만들어 txt로 저장한다.



<save 함수>

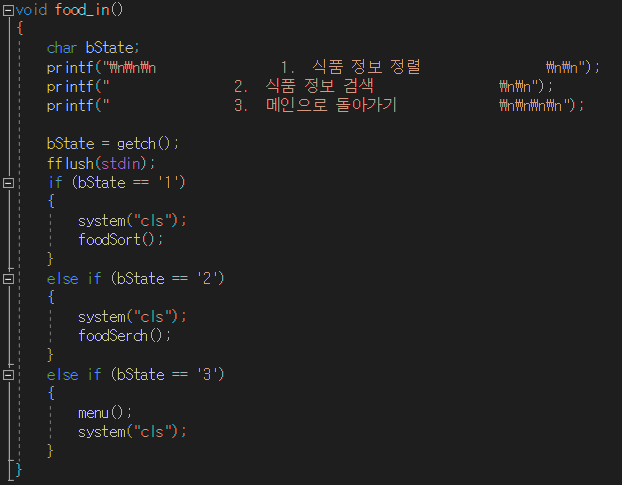
menu에서 3번을 입력할 경우, 등록해 놓았던 식품의 정보들을 냉장고의 저장해주는 함수이다.

즉, 식품 배송서비스에 등록되어 장바구니에 있던 식품들이 menu창에서 3번을 입력하면 냉장고에 저장된다. 냉장고에 들어있는 식품의 정보들을 출력하면 장바구니에 있던 식품들의 정보들이 나온다. 장바구니에 돌아가면 등록해보았던 정보들이 사라져있다.



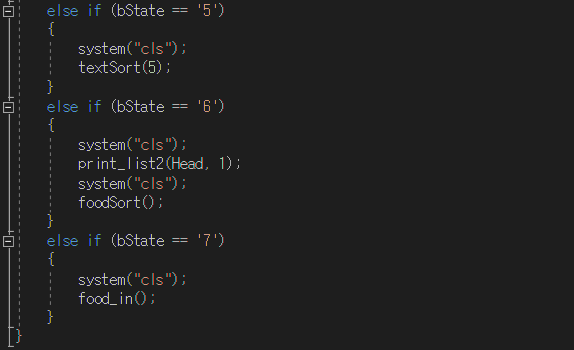
<insert 함수>

Insert는 함수는 저장한 식품의 정보에 동적메모리를 할당해 주는 함수이다. 예를 들어, 원하는 식품의 정보로 정렬을 할 때, 새로운 SLL을 만들어야 한다. 이때 메모리를 새로 할당하여 list를 만들 때 사용하는 것이다. 할당된 메모리가 없다면 기존의 list를 리턴해준다. 할당된 메모리가 있다면 연결 list가 처음이거나 마지막일때는 반복문을 탈출하고, 만약 아니라면 입력받은 kind와 sort의 값에 따라 if문에 들어가 연결 중 삽입할 곳을 찾는다. 찾은 삽입할 곳에 SLL를 이용해 data를 넣는다. 새로운 데이터가 들어간 list를 리턴해준다.



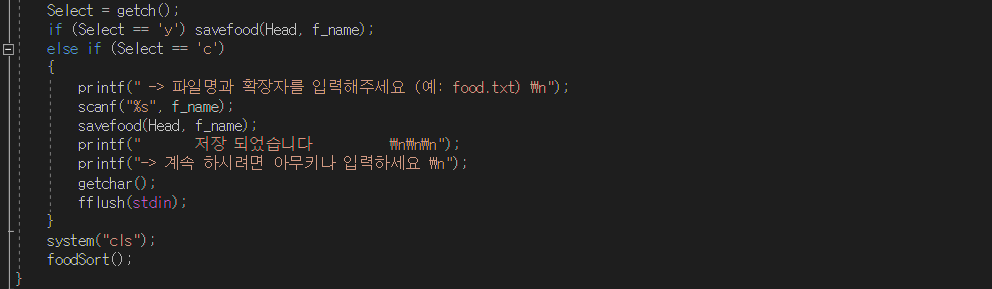
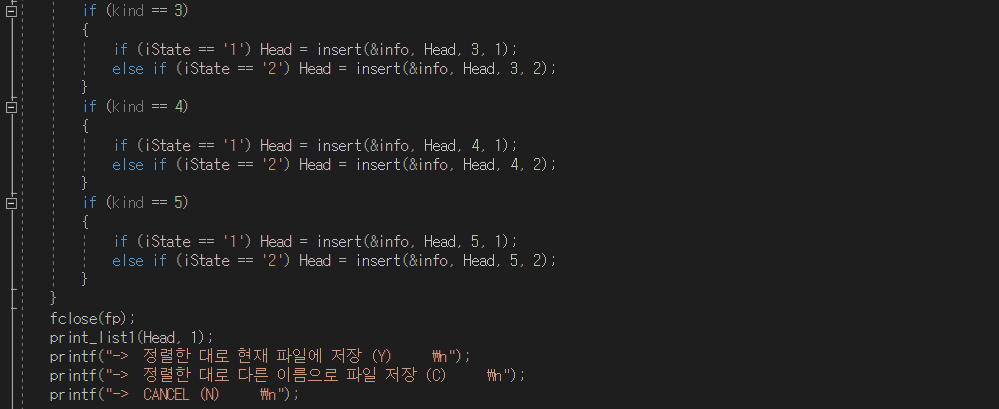
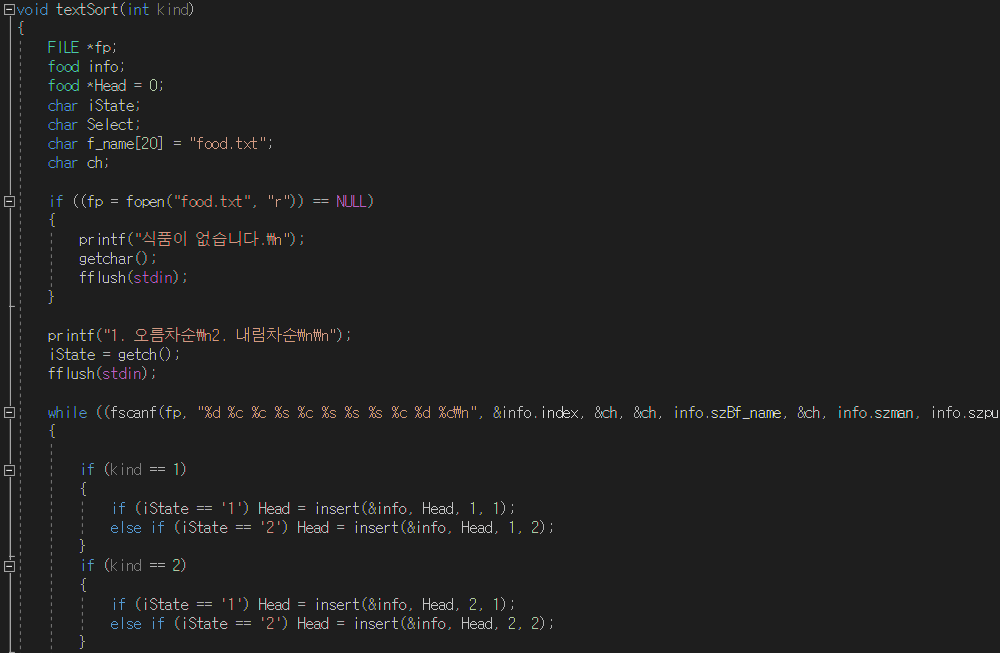
<food\_in 함수>

menu에서 6번을 누르면 검색 및 정렬 메뉴에 들어가도록 해주는 함수이다.



<sortselect 함수>

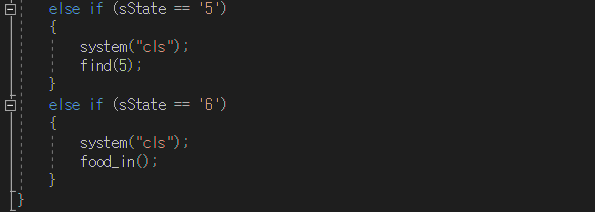
Food\_in함수에서 1번을 눌러 정렬을 선택한 후, 어떤 것으로 정렬을 할 지 정하는 함수이다. 1번부터 5번까지는 정렬 기준을 선택하는 것이고, 6번은 변경한 리스트를 따로 출력해볼 수 있고, 7번은 food\_in함수로 다시 되돌아가게 한다.



<sort 함수>

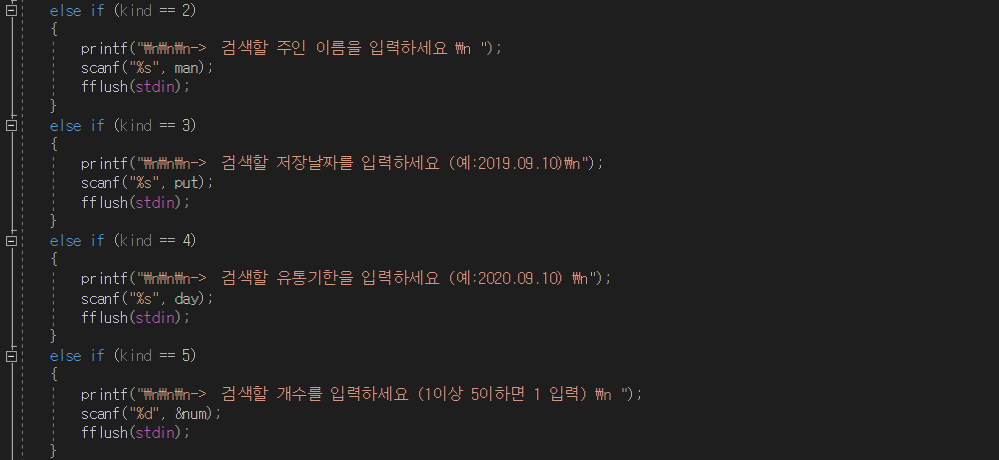
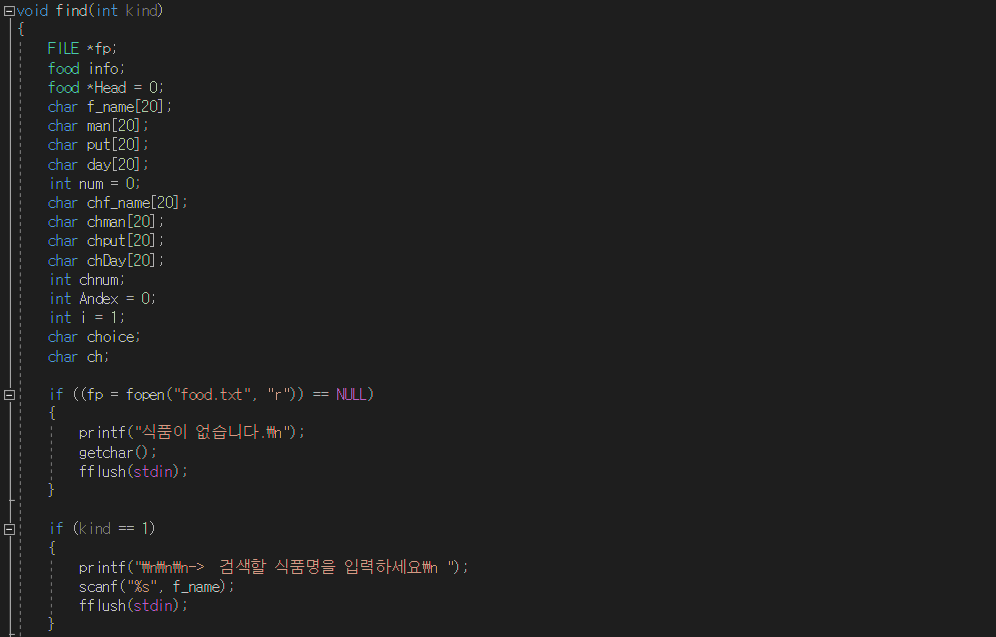
어떠한 기준으로 sorting할 지 결정했다면 그것을 오름차순으로 출력할지 내림차순으로 출력할지 입력을 받는다. 메모리를 할당해주는 insert함수를 이용하여 1번일 경우 오름차순으로 2번일 경우 내림차순으로 메모리를 할당한다. Print\_list1를 이용하여 정렬한 list를 출력한다.

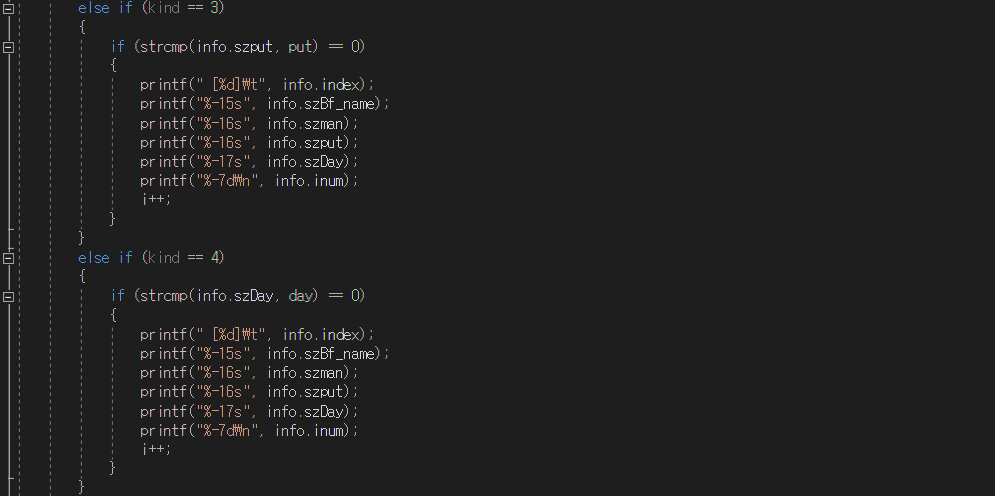
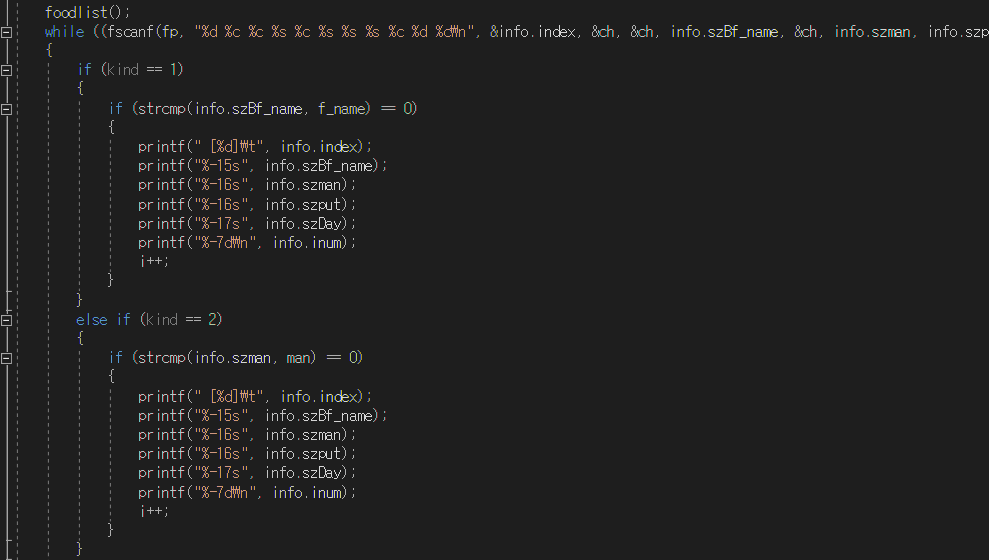
정렬한 식품 list를 저장하고 싶다면 savefood함수를 이용하여 txt파일로 저장한다.



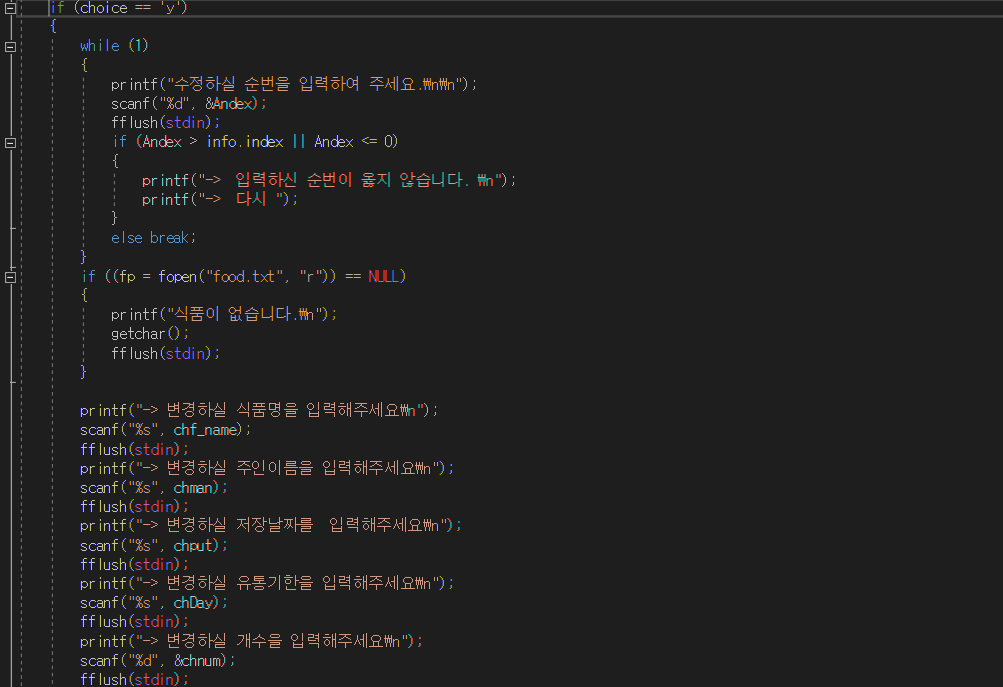
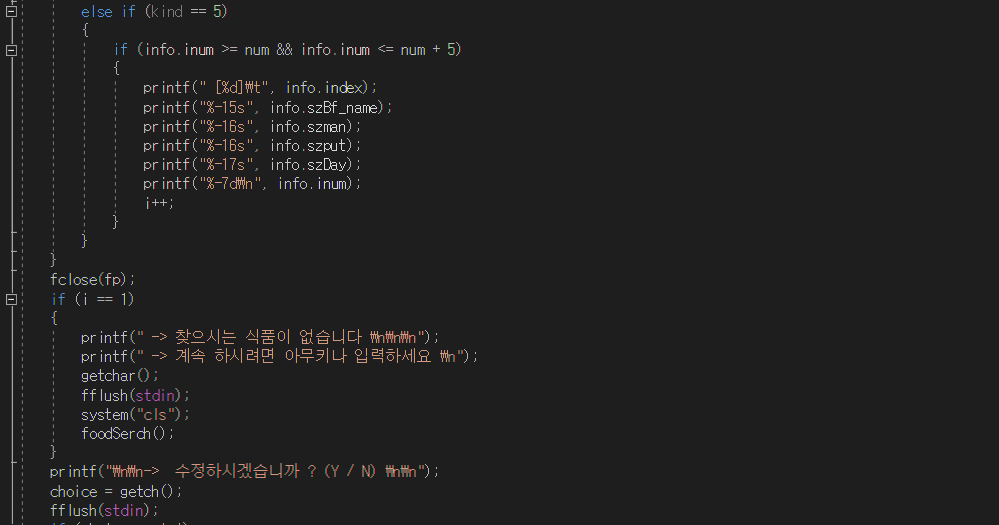
<search 함수>

menu에서 검색을 선택한 후, 정렬 및 검색에서 검색을 선택하면 실행되는 함수이다. Food\_in함수에서 2번을 눌러 정렬을 선택한 후, 어떤 것으로 정렬을 할 지 정하는 함수이다. 1번부터 5번까지는 검색 기준을 선택하는 것이고, 6번은 food\_in함수로 다시 되돌아가게 한다.

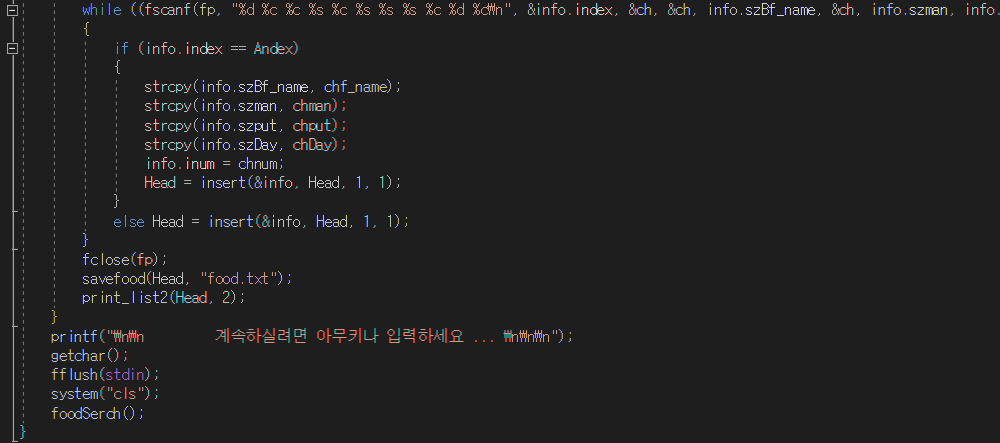
냉장고에 저장된 식품 list 중 원하는 식품의 data만 출력하기 위해 검색할 기준을 정한다.

(1번은 식품명, 2번은 주인 이름, 3번은 저장날짜, 4번은 유통기한, 5번은 개수) Strcmp를 이용하여 저장되있는 txt에 내가 입력한 것과 일치하면 일치한 식품의 정보를 출력해준다. 5번과 같의 개수에 대해서 출력할 때는 범위를 지정해서 그 안에 있는 식품의 정보를 출력해준다.

(아래 코드와 연결)

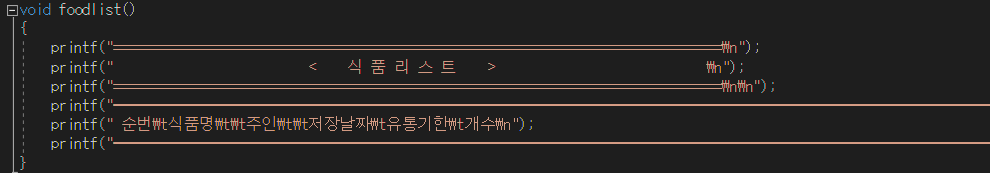


찾는 식품의 정보가 없을 경우는 다시 foodsearch함수로 되돌아간다. 만약 찾는 식품이 있다면 수정할 것인지 안할 것인지 정한다. 수정을 한다면 수정할 순번을 입력하여 그 순번에 있는 식품의 정보 변경한다. 오류를 예방하기 위해 입력한 순번이 현재 냉장고의 저장되어있는 index보다 크거나, 0보다 작거나 같을 경우, 다시 순번을 입력하도록 만든다.



<find 함수>

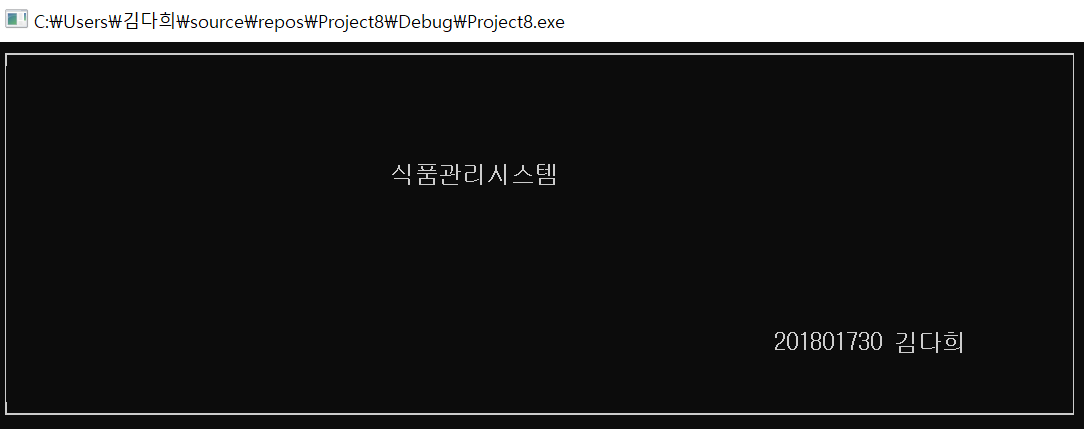
위에서 입력한 변경할 식품의 정보를 strcpy를 이용하여 카피한 후, insert 함수를 이용하여 새로운 정보를 삽입한다. 반복문을 이용하여 file를 읽어와 EOF를 만나지 않을 때까지 즉, 파일의 끝까지 갔을 때까지 반복한 후 탈출한다. Savefood 함수를 이용하여 수정한 데이터를 저장한 후, Print\_list2를 이용하여 수정한 식품의 정보를 출력한다.



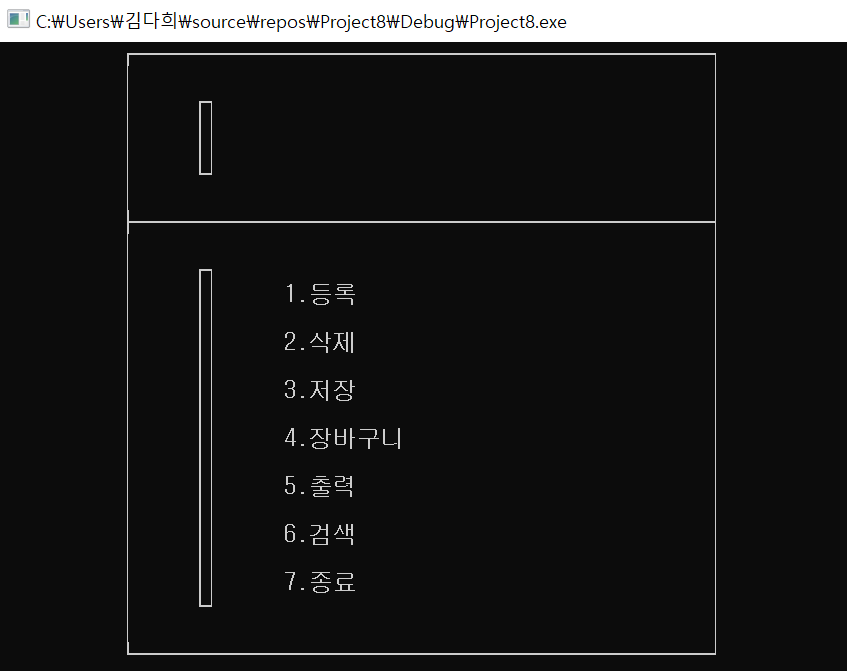
<foodlist 함수>

식품 list들을 출력할 때, 위에 표시할 때, 여러 번 출력하는 번거로움을 덜기 위한 함수이다.

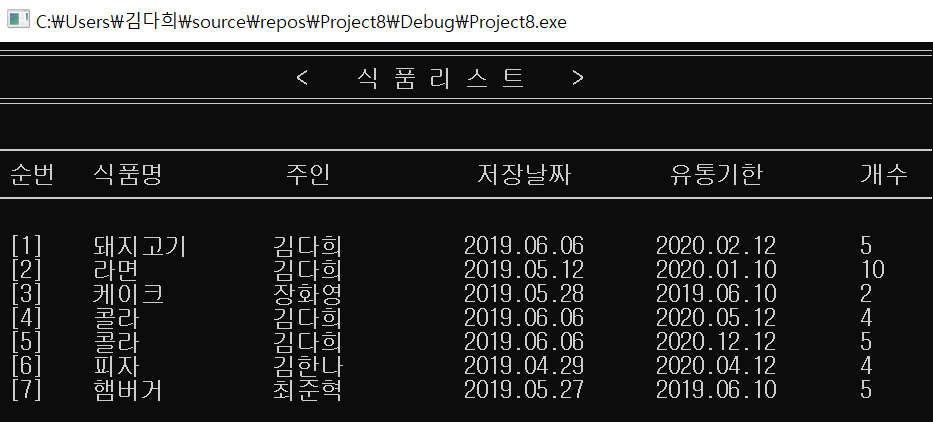
<결과>



<시작화면>



<menu 화면>



<기본 출력화면>



<종료 시 화면>