

0. 설계서 변경 사항

- 로그인 함수
- 사용자 예약하기 함수

1. 변경 된 기본 구조

- 예약 내역 txt파일은 정렬이 되어 있습니다.

1.전역 변수

- 사용자의 정보를 토큰으로 나눠 저장할 `vector<vector<string>>` 변수
 - 사용자의 정보 : 전화번호와 비밀번호
- 이용가능한 시작 날짜와 마감 날짜를 저장할 `int`형 변수
- txt파일의 제목이 될 이름들을 저장할 `vector<string>`
- 사용자들의 정보를 저장할 `vector<string>` 변수
 - 사용자들의 정보 : 텍스트 파일에 저장된 한 줄의 정보
- 불러온 건물, 방정보를 저장할 전역 변수 `vector<string>`

2.함수

- 시작 함수
 - 초기화 함수, 로그인 함수를 호출 한 뒤, 로그인 함수의 반환 값에 해당하는 함수를 호출합니다.
- 사용자 메뉴 함수
 - 예약하기와 예약 조회, 종료를 선택합니다.
- 사용자 예약하기 함수
 - 사용자는 이용하고자 하는 날짜를 입력합니다.
 - 입력받은 날짜가 예약이 가능한 범위 내의 입력인지 확인합니다.
 - 범위 내의 입력일 경우
 - txt 파일의 제목을 알려주는 함수를 호출해 해당 txt 파일을 불러오고, 불러온 정보를 token으로 나눈 후 `vector<vector<string>>`에 예약내역 정보를 저장합니다.
 - `vector<vector<string>>`에서 입력받은 날짜와 일치하는 곳의 시작 인덱스와 끝 인덱스를 변수에 저장합니다.
 - 범위 내의 입력이 아닐 경우
 - 날짜를 재입력받습니다.
 - 사용자는 해당 날짜의 이용하고자 하는 시작 시간 정보를 입력합니다.
 - 시작 시간을 입력받고, 이용하고자 하는 시간을 입력합니다.
 - 사용자는 시간을 정수 또는 .5의 형식으로 입력합니다.
 - 입력받은 시간이 사용 시간 범위 내의 입력인지를 확인합니다.
 - 입력받은 시간이 범위 내의 입력일 경우 예약 시간이 3시간이 넘지 않았는지 확인합니다.
 - 3시간이 넘었을 경우 사용자 메뉴로 돌아가게 됩니다.

- 사용자의 사용 시간 및 예약 시간 입력이 모든 조건 (사용 시간 범위 내, 예약시간 3시간 이내) 를 만족했을 때, vector<vector<string>>의 시작 인덱스와 끝 인덱스 사이에서 사용자의 희망 시간이 가능한 건물의 방을 찾아 출력합니다.
- 이 과정에서 방의 정보가 담긴 vector의 값을 미리 임시 vector에 저장한 뒤, vector<vector<string>>의 시작 인덱스와 끝 인덱스 사이의 값 중, 사용자의 희망 시간이 포함된 값이 있으면 해당하는 방을 임시 벡터에서 삭제해줍니다.
- 옮겨 담은 과정에서 @,\$,#이 들어간 값이 있는 경우는 임시 vector에 저장하지 않습니다.
- 임시 벡터가 비어있는지 확인한 뒤, 비어있다면 “예약할 수 있는 방이 없습니다” 라고 출력한 후 사용자 메뉴로 돌아갑니다.
- 임시 벡터가 비어있지 않다면, 임시 벡터에 있는 이용 가능한 방의 정보를 출력해 준 뒤, 사용자에게 건물과 층, 호실의 정보를 입력하도록 합니다.
- 입력받은 층이 존재하는 층인지 확인하고, 유효한 입력이 아닐 경우 재입력을 받습니다.
- 사용자가 입력한 층에 있는 방들이 해당하는 곳까지의 시작 인덱스와 마지막 인덱스를 갱신해줍니다.
- 반복문을 사용해서 예약내역과 겹치는 시간이 있으면 빨간색으로 전부 출력하고, 종료버튼도 같이 출력해줍니다. 아래 사진은 출력에 대한 예시이며, 설정된 건물 및 방 데이터에 따라 구현 화면과 차이가 있을 수 있습니다.

```

1. A101
  9.0 9.5 10.0 10.5 11.0 11.5 12.0 12.5 13.0 13.5 14.0 14.5 15.0 15.5
  16.0 16.5 17.0 17.5

2. B201
  9.0 9.5 10.0 10.5 11.0 11.5 12.0 12.5 13.0 13.5 14.0 14.5 15.0 15.5
  16.0 16.5 17.0 17.5

3. 종료

```

- 입력받은 방이 유효한 입력인지 확인하고, 유효한 입력이 아닐 경우 재입력을 받습니다. 이 때 종료에 해당하는 숫자를 입력했을 경우, 사용자 메뉴로 돌아갑니다.
- 입력받은 사용할 시간이 사용 가능한 시간인지 확인한 후, 예약 시간이 3시간 이하 인지 확인하고 예약을 성공시킵니다. 이 때 예약 시간이 3시간 이상일 경우, 사용자 메뉴로 돌아갑니다.
- 예약이 성공하면 예약내역 텍스트 파일에 갱신할 벡터에 해당 예약 내용을 추가해줍니다.
- 텍스트 파일을 WriteList()을 사용하여 내용을 갱신해줍니다.

- 사용자 예약내역 보기 함수

- 사용자가 입력한 날짜에 해당하는 txt 파일을 불러오는 함수를 호출합니다.
- 사용자의 전화번호와 일치하는 번호가 있는 줄을 출력합니다.
- 관리자 메뉴
 - 방 추가하기와 예약 확인하기, 종료를 선택합니다.
- 관리자 방 추가하기 함수
 - 관리자는 방을 추가하고자 하는 건물의 정보를 입력합니다.
 - 관리자가 입력한 건물의 정보가 방 vector에서 @가 포함된 값 중 있는지 확인합니다.
 - 만약 그 값이 없으면 다시 입력하도록 합니다.
 - 관리자는 추가할 방을 입력합니다.
 - 관리자가 입력한 건물에 총 방의 개수가 10개 이내인지, 같은 이름의 방이 없는지 확인하고 이 조건을 모두 만족하면 방을 추가합니다.
 - 만약 해당 건물의 모든 층의 방이 10개가 되면, 해당 건물 이름의 @값을 \$로 바꿔줍니다.
 - 만약 같은 이름의 방의 이름이 있다면 방의 정보를 다시 입력 받도록 합니다.
 - 처음 건물 정보가 저장된 벡터에 새로 추가된 내용을 넣습니다.
 - 추가된 내용은 건물이름+호실의 형태로 저장됩니다.(예_A301)
 - WriteList()함수를 호출합니다.
- 관리자 예약 확인하기 함수
 - 관리자가 입력한 날짜에 해당하는 txt파일의 제목을 찾아, 해당 txt파일의 내용을 출력해줍니다.
- 전체 프로그램 초기화(시작) 함수
 - 건물, 방정보 불러옵니다.
 - 불러온 건물, 방정보를 전역 vector<string>에 저장합니다.
 - 사용자 정보 불러옵니다.
 - 불러온 사용자 정보를 전역 vector<vector<string>>에 토큰 후 저장합니다.
 - 관리자에게 저장된 시작날짜에 +7을 끝나는 날까지 반복하고 string값을 벡터에 저장합니다.
 - 만약 이용가능한 마지막 날이, 마지막 일주일에 포함되지 않으면 마지막 날이 txt파일의 제목이 됩니다.
- 로그인 함수
 - 사용자는 전화번호와 비밀번호를 입력합니다.
 - 비밀번호는 입력하는 즉시 암호화 과정을 거쳐 암호로 변환됩니다. 이 때 암호화는 sha256 방식을 따릅니다.
 - 전화번호는 사용자의 정보가 담겨있는 vector<vector<string>>안에서 같은 것이 있는지 비교합니다.
 - 만약 전화번호가 같은 것이 있으면 사용자 vector<vector<string>>에 저장된 암호화된 비밀번호와 사용자가 입력한 비밀번호의 암호화 된 값을 비교합니다.
 - 비교된 값이 같으면 1을 반환합니다.
 - 입력한 정보가 관리자이면 2을 반환합니다.

- 전화번호가 같은 것이 없으면 `vector<vector<string>>`에 새 사용자의 정보를 저장합니다.
- 전역 선언된 `vector<string>`에 모든 사용자의 전화번호와 비밀번호를 합친 `string`을 저장하고 `WriteList()`함수를 호출합니다.
- 이후 3을 반환합니다.
- 사용자가 입력한 날짜가 포함되는 txt파일을 찾는 함수
 - 날짜가 저장된 벡터를 기준으로 사용자 입력 날짜가 해당된 `string`을 찾습니다.
- txt파일을 여는 함수

```
bool file_open(string f_name,vector<string>v)
{
    파일열고
    if(성공)
        vector<string>에 line단위로 저장
        return true;
    else
        return false;
}
```

- txt파일을 열기 위한 함수는 인자로 파일이름을 넘겨준 후, 파일 open성공 여부에 따라 `return`값을 반환해줍니다.
- txt파일을 열기에 성공했을 때, 인자로 넘겨준 `vector`에 line단위로 `string`을 저장합니다.
- txt파일에 열기에 실패했을 때는, `false`를 반환합니다.

- txt파일의 내용을 출력하는 함수

```
void PrintList(vector<string>v)
{
    for(벡터의 크기만큼)
        cout으로 출력
}
```

- txt파일을 읽은 후 `vector`에 저장된 `string`값을 출력해주는 함수입니다.
- txt파일의 내용을 공백으로 분리해주는 함수

```
제목 없음 - 메모장
파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)
vector<vector<string>> token(vector<string>,int size){
    //인자로 받은 벡터는 텍스트 파일의 한 줄의 데이터들이 저장되어 있다.
    //인자의 size는 한 줄을 구성하고 있는 요소의 개수를 의미한다.
    한 줄을 담을 새 vector1 생성
    for(벡터 크기만큼){
        {
            한 줄의 요소를 나눠서 담을 새 vector2 생성
            공백을 기준으로 size만큼 분리하고 새 vector2에 저장
        }
        vector2를 벡터 vector1에 담는다.
    }
    return vector1
}
```

- 변경된 예약 내용 저장하는 함수

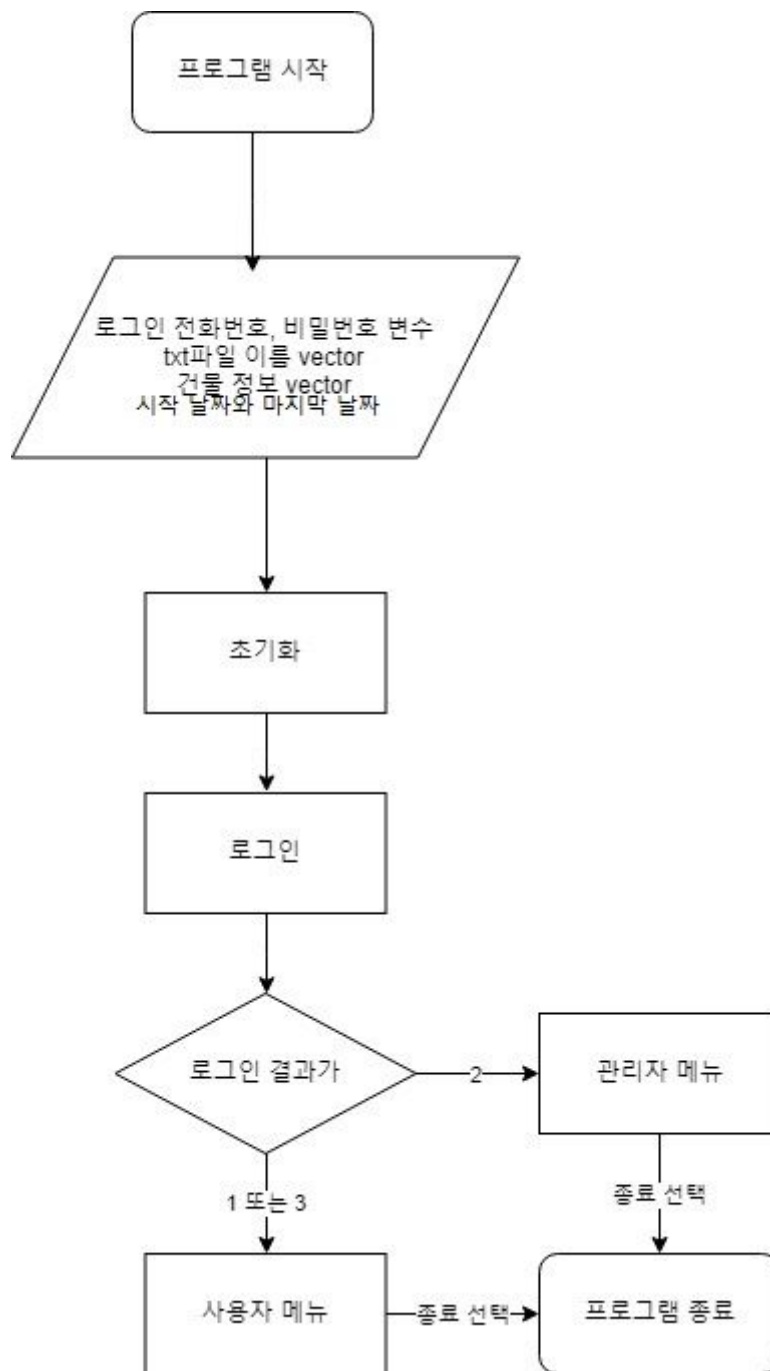
```
void WriteList(vector<string>,string 파일이름)
{
    vector를 sort
    변경된 모든 데이터를 갖고 있는 vector를 파일에 덮어 쓴다.
}
```

- 파일이름을 넘겨주고 해당되는 파일을 열고 인자로 받은 벡터로 파일의 내용을 덮어씁니다.

2. 순서도

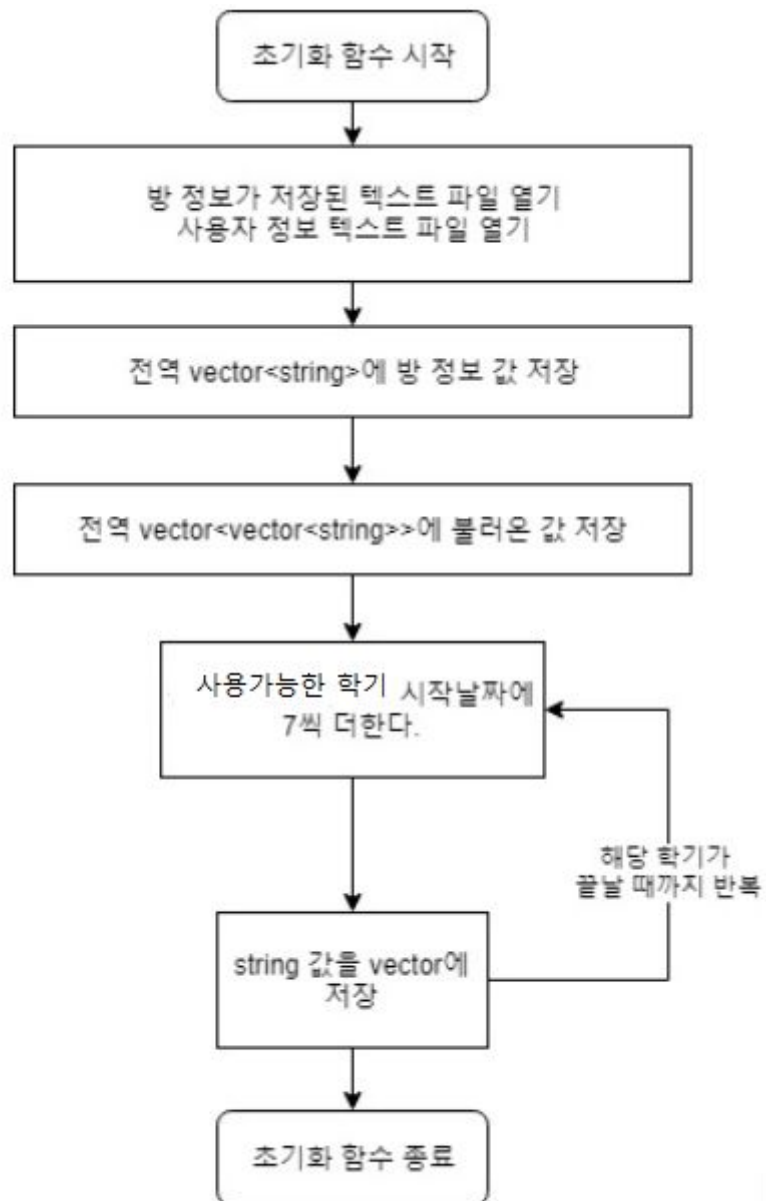
- 순서도에 적힌 변수는 특별한 경우를 제외하고는 그 부분에서 생성합니다.
- 모든 실행을 마친 뒤, 본래의 메뉴로 돌아갈 때, q를 누르면 돌아가도록 합니다.

2.1 프로그램 전체 순서도

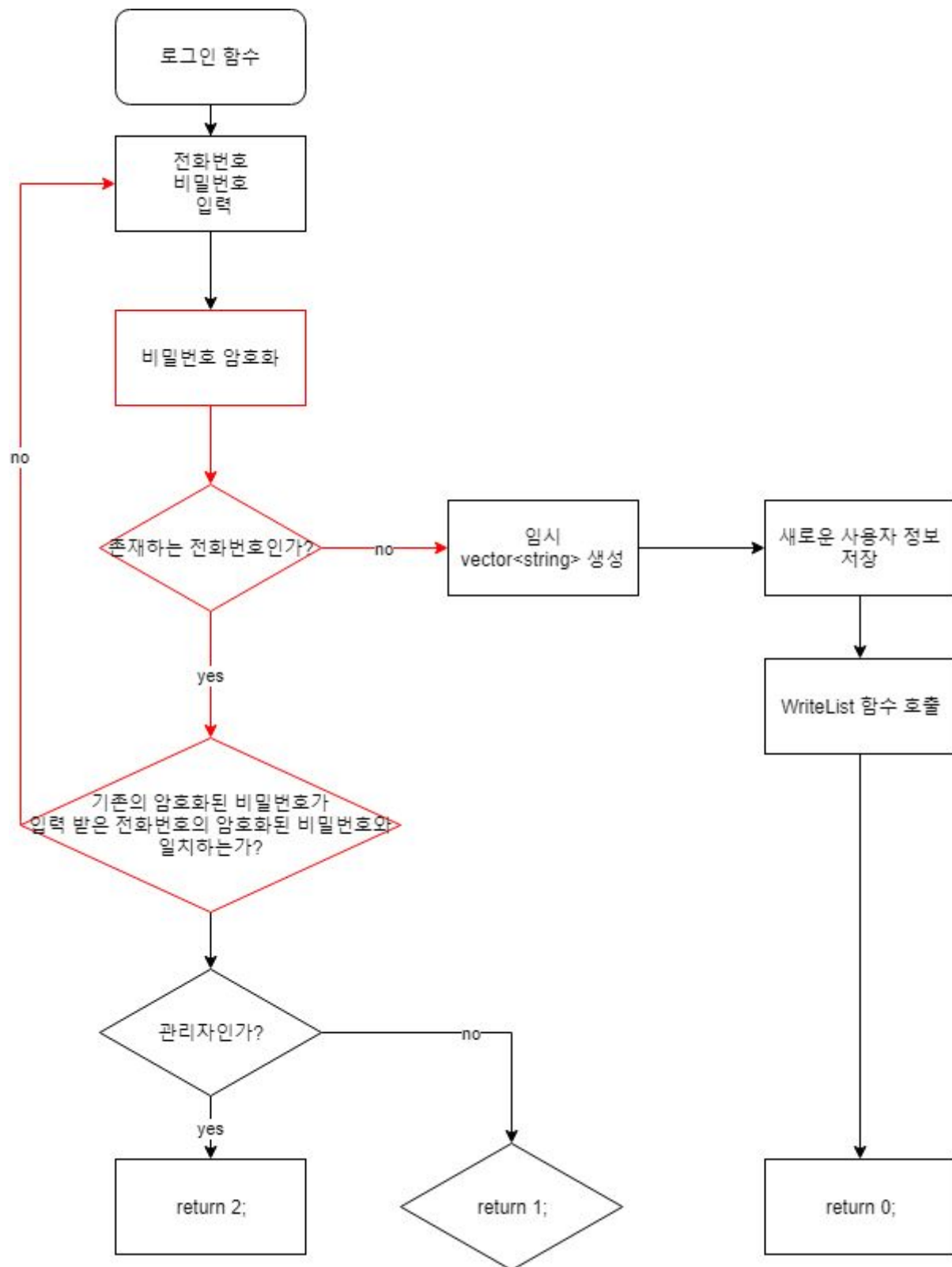


2.2 프로그램 내 함수 순서도

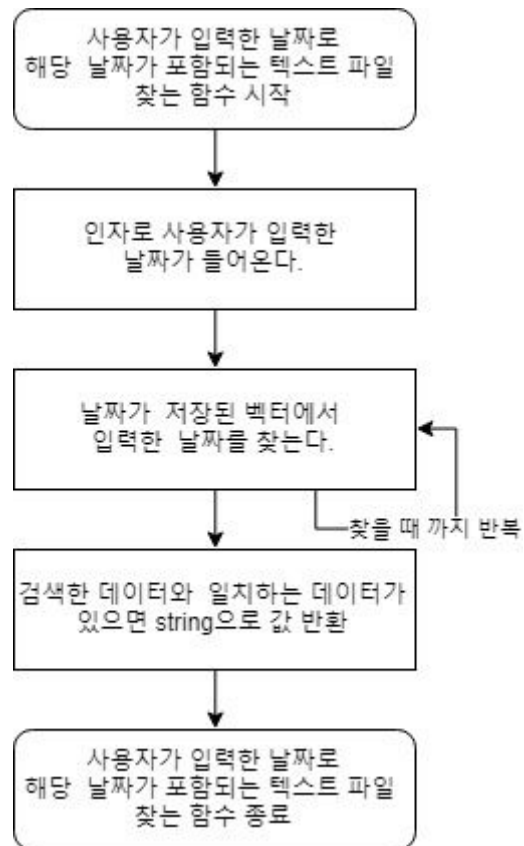
- 초기화 함수



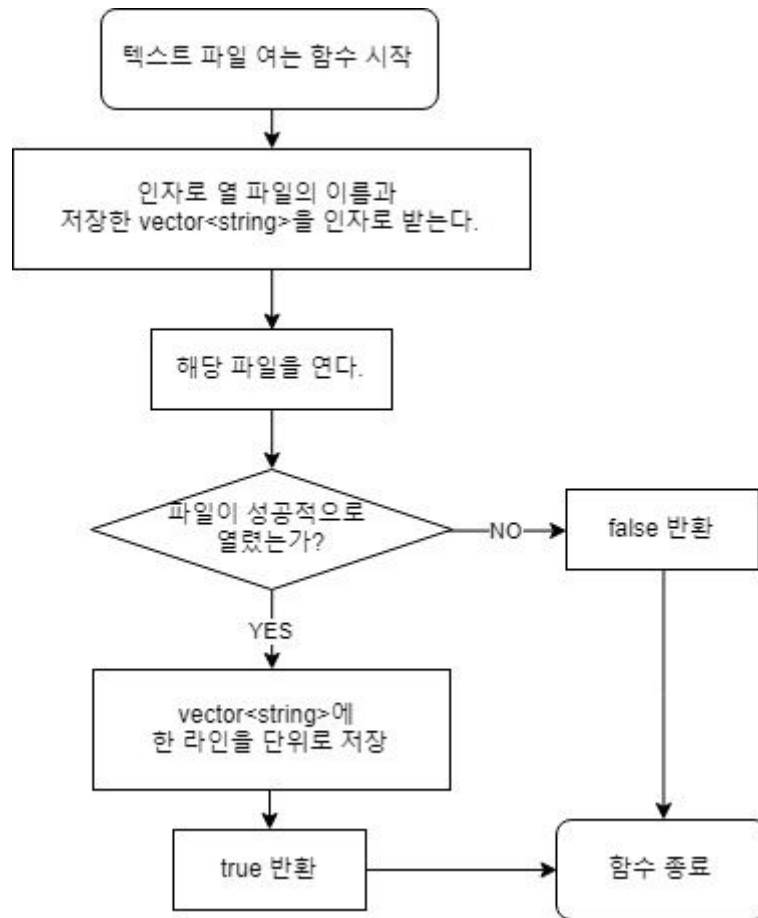
- 로그인 함수



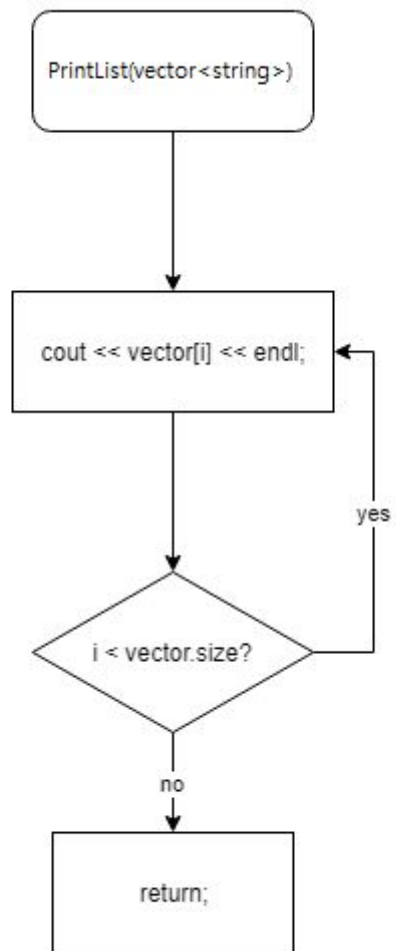
- search 함수
 - 날짜를 입력한 경우, 그 날짜가 범위 내에 있는지 확인해줍니다.



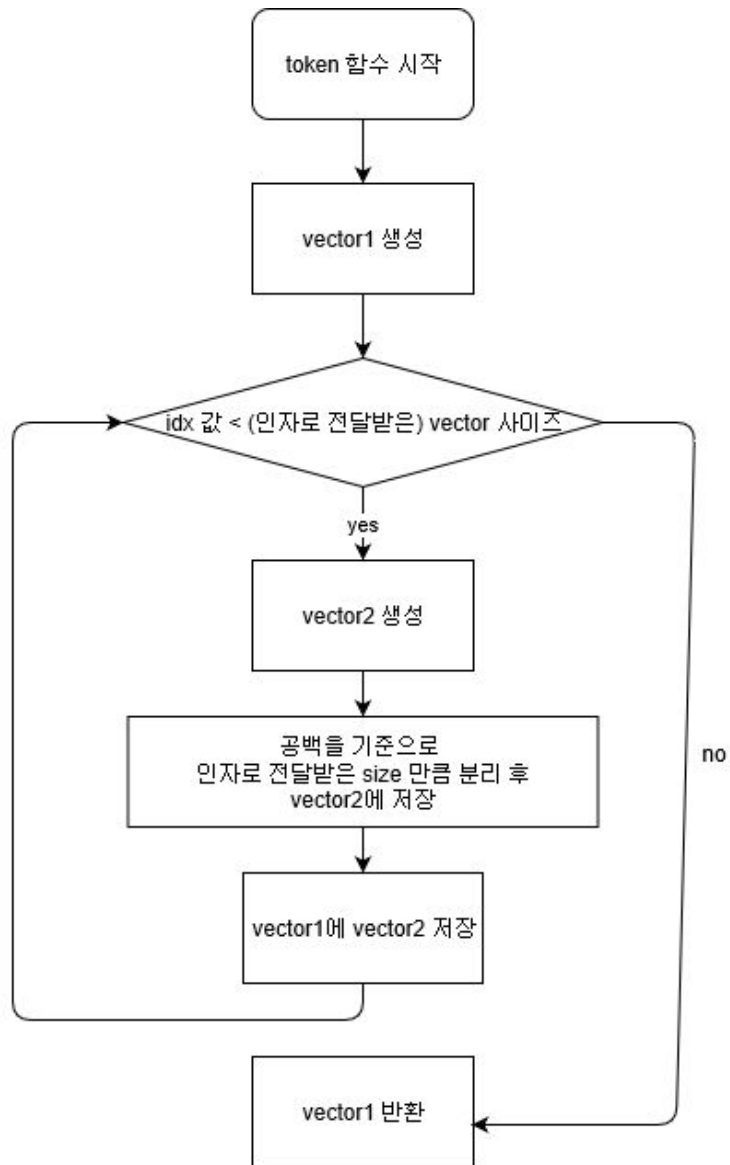
- `openfile` 함수



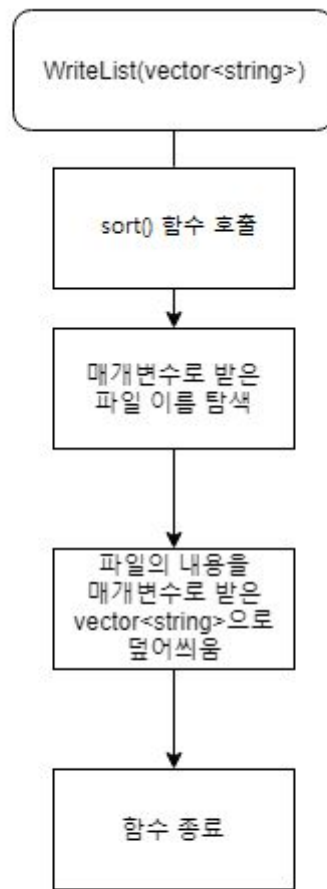
- print 함수



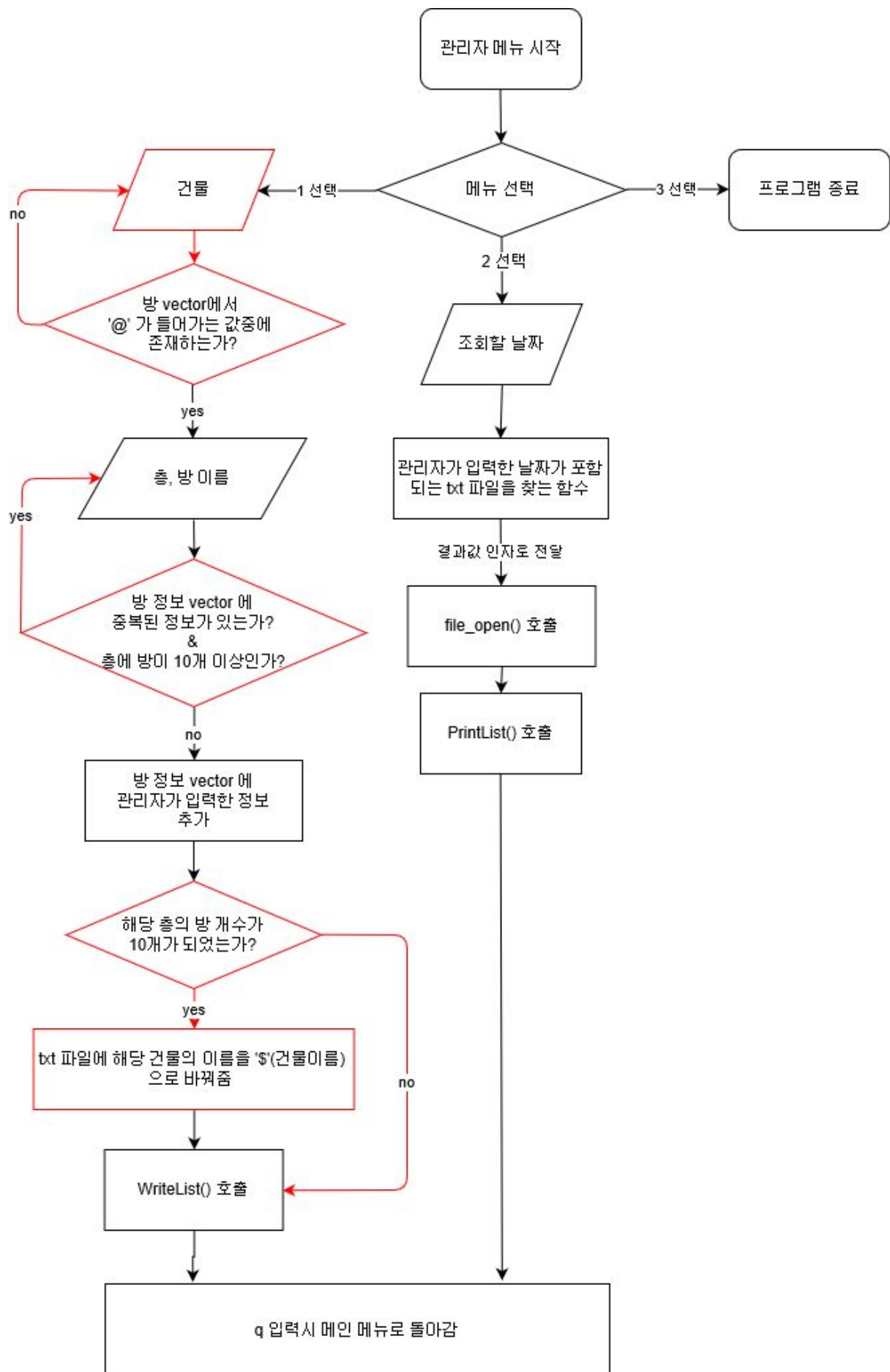
- token 함수

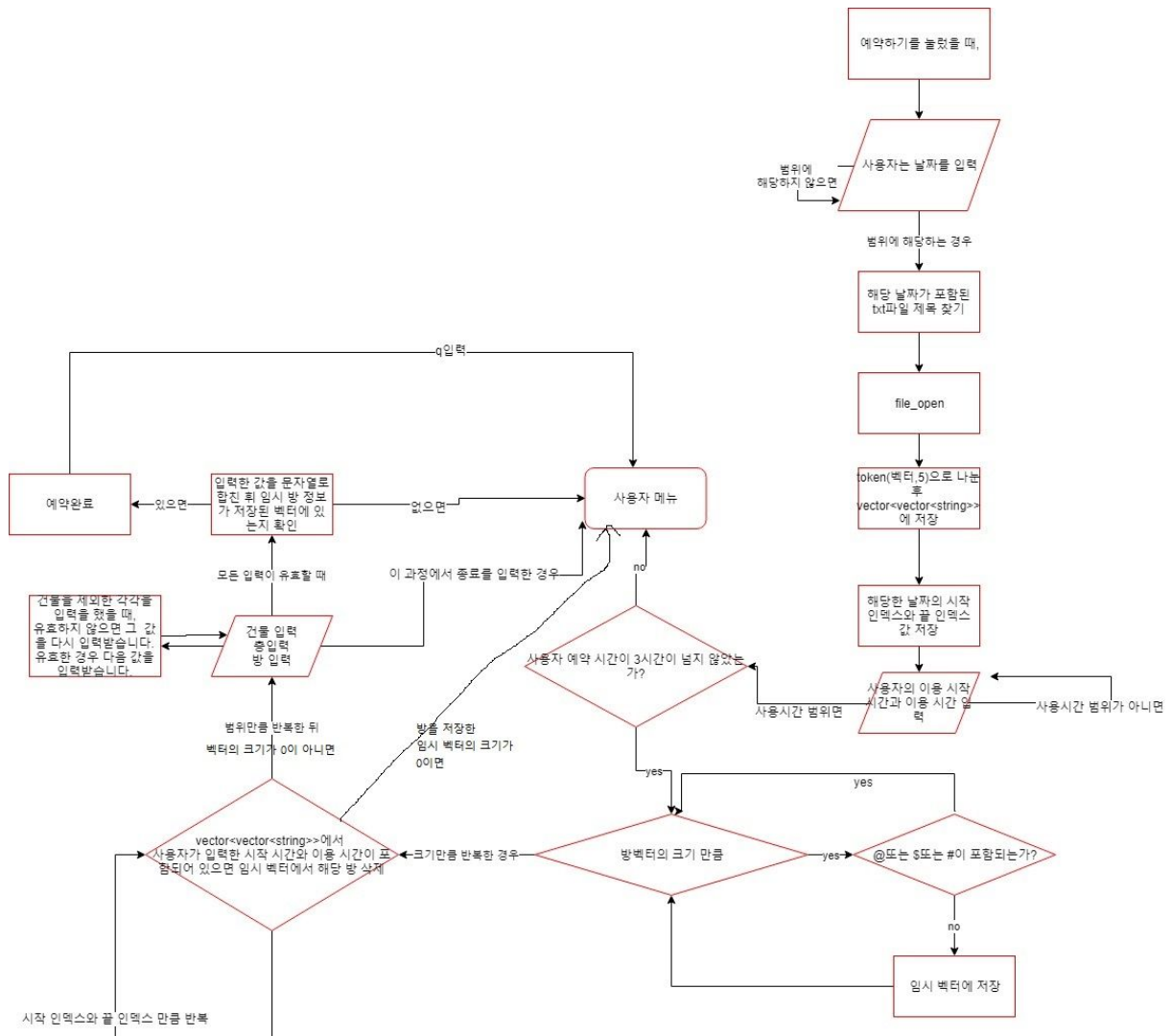


- WriteList 함수



- 관리자 메뉴 함수





* 위의 순서도에서 건물, 층, 호실 정보 읽어오는 loop만 아래와 같이 수정합니다.

