

## CSE3081 프로그래밍 숙제#0 (Due: 10/2 일, 오후 10시까지 email 제출)

수업에서 다룬 linked list를 이용하여, 그래프에서 Euler path 또는 cycle이 존재할 경우 이를 찾아 출력하는 프로그램을 만들어 보자.

과제 작성 및 제출 요령은 다음과 같다(아주 자세히 읽어야 한다)

가. 인터넷에 게시한 파일들을 VS2012에 우측 그림과 같이 등록한 console application 프로젝트를 생성한다.

나. 학생들이 작성하여야 할 파일은 snnnnnn\_phw00.cpp이므로, 학생들은 이 파일의 이름에서 nnnnnn을 자신의 학번 뒤 6자리로 파일 이름을 바꾸어 등록해야 한다.

다. 입력된 그래프는 adjacency matrix이다. 만일 정점 v와 w간에 에지가 2 개 있으면  $G[v][w] = G[w][v] = 2$ 이다. 이를 파일에서 읽어 2차원 변수 G에 저장되어 있다.

라. 각 함수 설명은 다음과 같다.

```
int getOddDegreeNum(int **G, int N, int *start)
```

이 함수는 G의 각 정점의 degree를 조사하여 홀수 degree를 갖는 정점의 개수를 반환하는 함수이다. 아울러, 만일 홀수 degree를 갖는 정점의 개수가 정확히 2라면 start에 홀수 degree 정점의 index 하나를 저장하여야 한다. G는 2 차원 배열이고, N은 배열의 크기이다. 쉬운 함수이니 금방 만들 수 있다.

```
void Find_a_Path(int **G, int N, int start, ptr_L **path, ptr_L **path_end)
```

이 함수는 정점 start ( $=0,1,...,N-1$  중 하나) 부터 시작하여 더 이상 진행할 수 없을 때까지 경로를 계속 찾아 그 경로를 singly linked list로 만들어 그 시작을 path에 그리고 끝을 path\_end에 저장하여 Find\_Euler함수에 제공하는 기능을 수행한다. 경로를 찾는 과정에서 지나간 에지는 G에서 지워야 한다. 즉, 에지 (v,w)를 지났다면  $G[v][w]$ 와  $G[w][v]$  값을 하나씩 감소해야 한다. Linked list element의 정보는 vertex index이다(0,1,...,N-1). 그리고, path와 path\_end는 ptr\_L 포인터이다.

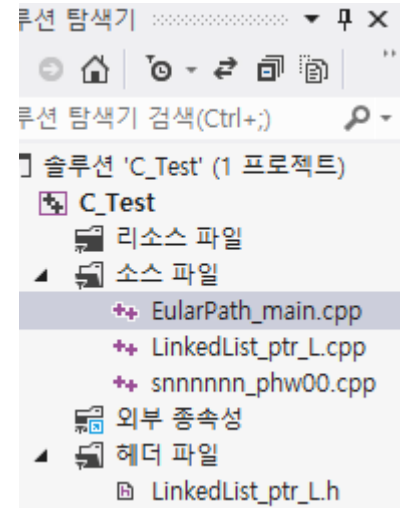
```
ptr_L *Find_Euler(int **G, int N, int start)
```

이 함수는 정점 start에서 시작하여 start로 끝나는(Euler cycle) 또는 다른 정점에서 끝나는(Euler path)를 찾아 이를 singly linked list에 저장하고 이의 head를 반환하는 함수이다. 수업시간에 설명한 것과 같이 일단 start에서 시작하는 경로 P를 하나 찾고, 이 경로를 스캔하여 에지가 연결된 정점 v가 있을 경우, v에서 시작하여 v로 끝나는 경로 Q를 찾아서, P에 Q를 끼워 넣는 과정을 더 이상 에지가 없을 때까지 같은 과정을 반복하면 된다. 이 끼워 넣는 작업을 singly linked list로도 충분히 가능하다. 어찌 할 수 있는지 생각해 보자.

마. 게시한 E\_in.txt는 학생들의 프로그램을 테스트하기 위하여 주어진 예이다. 이를 사용하여 자신이 작성한 프로그램을 debug해 보자. 프로그램은 redirection을 통해 수행하여야 한다. 즉, 명령 프롬프트에서 다음과 같이 입력하여 프로그램을 수행한다. 실행 파일을 Euler.exe라고 하자.

```
C:\WEuler.exe < E_in.txt > E_out.txt
```

이러면, E\_out.txt에 찾은 경로가 저장될 것이다.



- 바. Euler\_Finding.exe 파일은 샘플 실행 프로그램이다. 이를 수행하여 입출력이 어찌 되는지 알아보고 나아가 자신의 프로그램 결과와 비교해 보자(담당 교수가 만들었으므로 오류가 있을 수 있다. 이 경우 무슨 오류인지 알려주면 수정하여 다시 게시할 것이다).
- 사. Graph\_Gen\_E.exe는 입력 데이터를 만들어주는 프로그램이다. 이를 수행하여 큰 그래프 입력을 만들어 실행해보자. 실제 채점에서는 아주 큰 데이터를 만들어 시험할 것이다.
- 아. 자신의 프로그램이 맞는지 조사하는 것은 Euler\_Finding.exe를 실행하여 저장된 파일과, 자신이 생성한 파일을 비교하여 알 수 있다. 명령 프롬프트에서 FC라는 명령은 두 파일을 비교해 준다. **FC /?**라고 입력하면 도움말이 나오니 이를 참조해서 비교해 보자.
- 자. 이 과제는 linked list에 익숙해 지는데 상당한 도움이 되니 잘 수행해 보자. 과제는 일찍 마쳐야 한다. 마감 바로 다음 주에 또 다른 과제 마감이 기다릴 수 있다. 과제 수행 중 질문이 있으면 인터넷 카페에 글을 올리자. 가능한 답을 줄 것이다.
- 차. **과제 제출은 snnnnnn\_phw00.cpp 만을 email에 첨부**하여(반드시 첨부) 제출한다. Email 주소는 [2016algorithm@naver.com](mailto:2016algorithm@naver.com)이다. 프로젝트 전체를 보내지 말자(감점함)
- 카. 메일 제목은 "알고리즘 PHW00 학번 이름"을 적고 내용은 아무것도 적지 않는다.
- 타. 파일 이름에 오류가 있으면 감점한다. **snnnnnn\_phw00.cpp**외에는 아무런 파일도 수정하지 않는다. 채점 시 만일 컴파일 오류 또는 링크 오류가 발생하면 이유 불문 점수를 받을 수 없다.
- 파. 파일을 여러 번 보내면 안되고, 단 한 번만 보내야 한다. 특별한 이유 없이 두 번 이상 보내면 감점할 수 있다.