**Graph ADT**

**2013190250 이동현**

Graph 구조체 설정

1. GRAPH – 그래프. 세가지 변수로 구성. 1. 노드 개수. 2. 첫번째 노드 연결 3. 값 비교를 위한 포인터 함수.
2. VERTEX – 노드. 여섯가지 변수로 구성. 1.다음 노드 연결. 2. 데이터 3. 들어오는 arc 갯수 4. 나가는 arc 개수. 5. Processed, 6. Arc에 대한 연결.

Processed는 traversal 함수에서 이용된다.

1. Arc – 선. 세가지 변수로 구성. 1. 연결된 노드. 2. 무게. 3. 다음 arc 연결.

Graphcreate – 포인터 함수를 매개 변수로 받는다. 메모리를 새로 할당하여 그 구성값을 초기화하여 graph 구조체를 반납한다.

graphInsVrtx – 대상 그래프와 데이터를 매개변수로 받음. 메모리를 할당하여 newPtr의 각 값을 초기화하고 그래프 구조체에 저장되어있는 포인터함수를 통해 매개변수로 받은 값과 그래프 각 노드의 값을 비교하여 적정한 자리에 노드를 추가 입력함.

graphDltVrtx – 대상 그래프와 키 데이터를 매개 변수로 받음. 데이터 값을 기준으로 일치하는 노드를 찾고 그 노드가 들어오는 arc와 나가는 arc가 없을 경우 제거함.

graphInsArc – 대상그래프, 출발 키, 도착키, 입력할 선의 무게를 매개변수로 받는다. 출발할 값과 노드의 값과 일치하는 노드를찾고 있다면 도착 키와 같은 값을 가지는 노드를 찾는다. 둘다 발견될 경우 입력한 선의 무게과 함께 arc로 연결한다.

graphDltArc – 마찬가지로 출발키와 도착키를 기준으로 일치하는 값을 찾을 경우 arc를 지운다.

graphEmpty – graph의 노드 개수가 0일 경우 참, 아닐 경우 거짓

graphFull – 메모리 할당에 실패할 경우 참, 아닐 경우 거짓

graphCount – graph의 노드 개수를 반납.

graphRetrVrtx – 그래프에서 pkey와 일치하는 노드를 찾고 그 노드의 데이터를 dDataOut 이중포인터로 가리킴.

graphDpthFrst – 스택을 이용한 traversal. 모든 노드의 processed를 0으로 셋팅함.

이후 각 노드에 연결된 노드들의 proessed가 1보다 작다면, 즉 0이라면 스택에 넣는다. 이후 스택에서 차례차례 팝스택을 하는데, 팝을 하면서 그 노드의 processed는 2로 변경한다. 즉 노드가 한번 확인되었음을 기록한다.

graphBrdFrst - 큐를 이용한 traversal. 이하 graphDpthFrst의 과정과 같음.