



2017 Dev# C++ Study

진행자: 박태욱

진행 순서

1. 바뀐 C++스터디 구성시간 소개
2. 모임 소개, 인원 모집
3. 프로젝트소개 – 지난주 ppt 참고할 것
4. 지난 주 개발 상태
5. 링크드리스트란?
 - 상속이란?
6. 도전과제

1.바뀐 C++ 구성시간 소개

1. 앞의 1시간은 여전히 발표로 이루어질 예정. 발표는 과제/모임/모임성과 발표회 등으로 이루어짐.
2. 스터디 시간에 모든걸 해결하는 것을 목표로 할 예정. 따라서 실전 코딩 시간의 비중이 크게 늘어남
3. 과제 수행 여부는 맘대로, 과제 코드를 바탕으로 발표 지원자들에게는 다음과 같은 혜택이 있습니다.
 - 1대1 피드백 시간을 통해
 - 좋은 코딩 습관 소개
 - 생산성을 높이는 코딩
 - 스터디 시간에 소개하지 못한 내용 소개
 - 잘못 알고 있던 지식을 바로잡을 수 있음
4. 여러가지 발표 -> 스터디 공지 발표 -> C++ 주제 내용 소개 및 실전코딩 반복

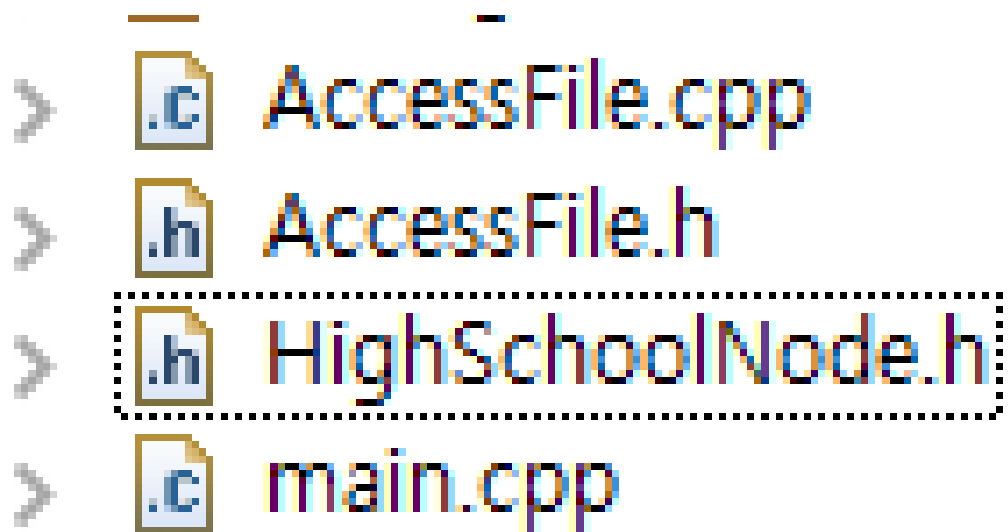
2. 모임 소개 & 인원 모집

1. 안드로이드 모임- 기획자 : 박정민
2. 리눅스 모임 – 기획자 : 장종원
3. C++++모임 – 기획자 : 박태욱
4. 건강 모임 – 기획자 : 전인호

4. 지난 주 개발 상태

지난 주 목표 – 파일에서 한 줄을 읽은 후 노드를 구성하는 것.

노드 : 링크드리스트를 구성하는 정보객체 하나



- HighSchoolNode 구조체 – 파일 한 줄의 정보를 저장하는 노드
- AccessFile 클래스 – 파일에서 정보를 읽어오는 기능의 클래스
- Main.cpp – 위의 두 가지 코드 테스트용 메인 코드

5. LinkedList란?

다양한 함수를 알아보자

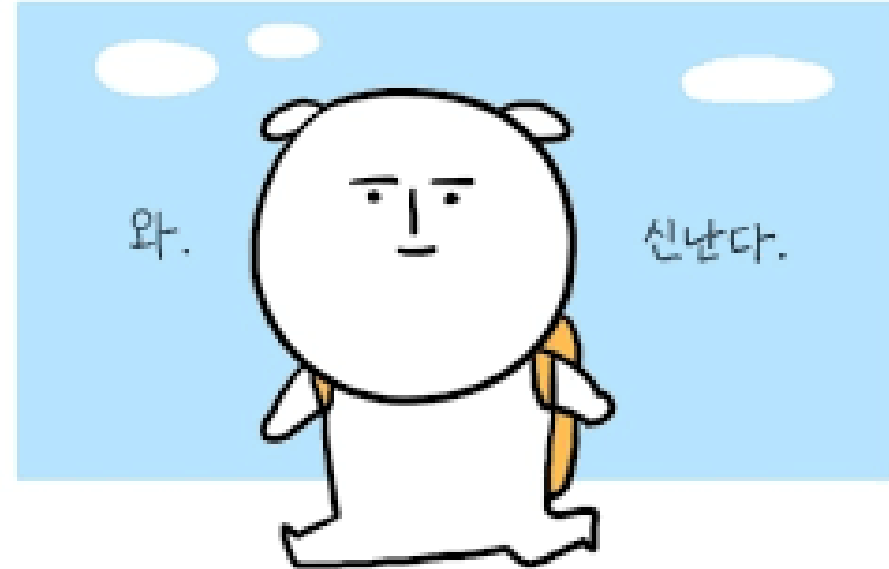
1. `LinkedList();`(생성자)
2. `~LinkedList();`(소멸자)
3. `Void add_ahead(const HighSchoolNode * data);`
4. `Void add_before_tail(const HighSchoolNode * data);`

5. LinkedList란?- 상속에 대해 알아보자

상속 – 기존의 클래스의 기능을 바탕으로 새로운 클래스를 작성하는 방법

상속에는 **public 상속**

상속 형태 Base 클래스	public 상속
public 멤버	public
Protected 멤버	protected
Private 멤버	접근 불가



학교에 다니고 있다는 것이다...

5. LinkedList란?- 상속에 대해 알아보자

AccessFile 함수를 public으로 상속받은 후 3, 4번 함수를 개량하자.

*3. Void add_ahead(const char * chp);

*4. Void add_before_tail(const char * chp);

5. LinkedList란?

다양한 함수를 알아보자

5. bool search(string & data, HighSchoolNode & getData);

6. Bool delete_element_with_string(string & data);

7. Void display();

8. sort(bool (*compare)(const HighSchoolNode &data1, const HighSchoolNode & data2)); //메인에는 테스트코드 없음

5. LinkedList란?

다양한 함수를 알아보자

7. Void display();

8. sort(bool (*compare)(const HighSchoolNode &data1, const HighSchoolNode & data2));

9. makeFile(const string & filename);

5. 도전과제

다음 함수에 도전해보자.

1. `bool search_multi(const string & data, vector<HighSchoolNode> &getDatas);`
2. `Bool delete_element_with_string_multi(const string & data);`
3. 기존의 `search` 함수는 무조건 소재지 검색만 가능한데, 이를 학교명, 학급수에 따라 검색이 가능하도록 함수를 개량해보자.
4. `HighSchoolNode`에 총인원수, 교실 수 멤버를 추가한 후 노드를 생성할때마다 멤버의 값을 계산하도록 `makeNodeInfo` 함수를 바꾸자. 그 후 `search` 함수가 총인원수, 교실 수 멤버를 이용해 검색이 가능하도록 `searc` 함수를 개량해보자.