C++ 파일 입출력, 예외 처리와 C언어와의 링크 지정

202004103 정원석, 202004904 이종하

<C++ 파일 입출력>

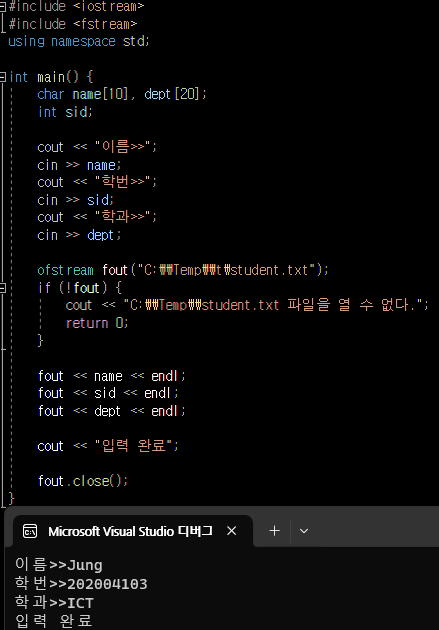
텍스트 파일 – 글자 혹은 문자들로 구성되는 파일  
\n, \t 등의 특수 문자도 포함하고 문자마다 문자 코드(이진수)를 할당되어 있다.  
파일의 종류로는 txt 파일, HTML 파일, XML 파일, C++ 소스 파일 등이 있다.

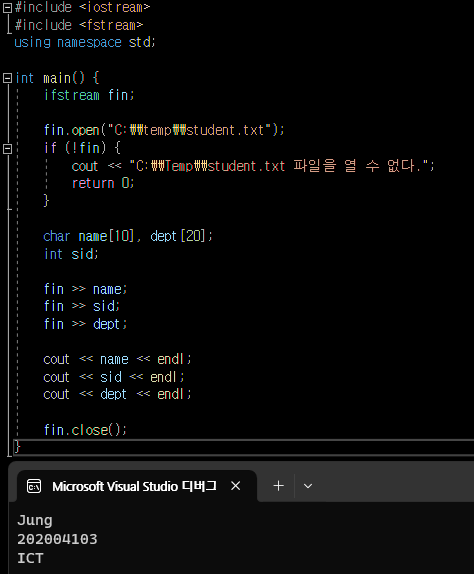
바이너리 파일 – 문자로 표현되지 않는 바이너리 데이터가 기록된 파일  
바이너리 파일의 각 바이트는 문자로 해석되지 않는 것도 있어 파일을 만든 응용프로그램 만이 해석 가능하다.  
바이너리 파일의 종류로는 jpeg, bmp등의 이미지 파일, mp3 등의 오디오 파일, hwp, doc, ppt 등의 확장자를 가지 멀티미디어 문서 파일, obj, exe등의 확장자를 가진 컴파일된 코드나 실행 파일이 있다.

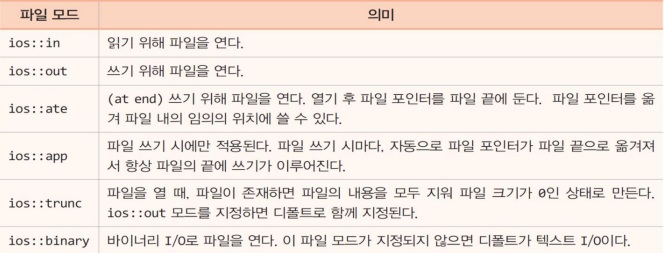
C++ 파일 입출력 라이브러리를 사용하고 <fstream> 헤더파일과 std 이름 공간의 선언이 필요하다.

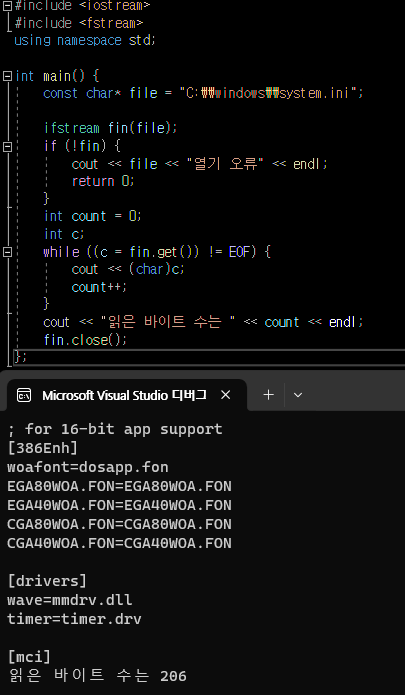
파일 입출력 방식은 텍스트 I/O와 바이너리 I/O의 두 방식이 있는데 텍스트는 문자 단위로 파일에 쓰거나 읽기에 사용되고 텍스트 파일에만 적용된다. 바이너리는 바이트 단위로 파일에 쓰거나 읽고 텍스트 파일과 바이너리 파일 모두 입출력이 가능하다.

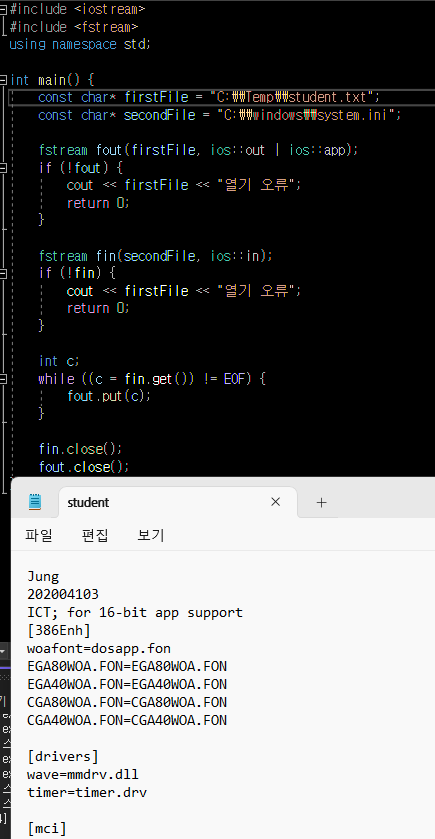
예제 12-1)



예제 12-2)  


파일 모드 – 파일 입출력에 대한 구체적인 작업 형태에 대한 지정이다.  


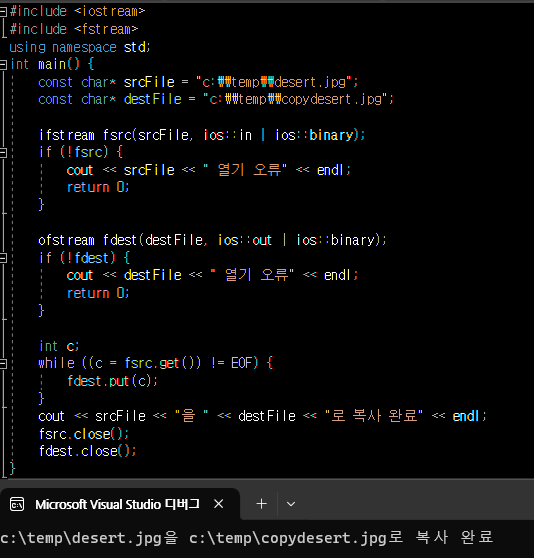
예제 12-3)  


예제 12-4)  


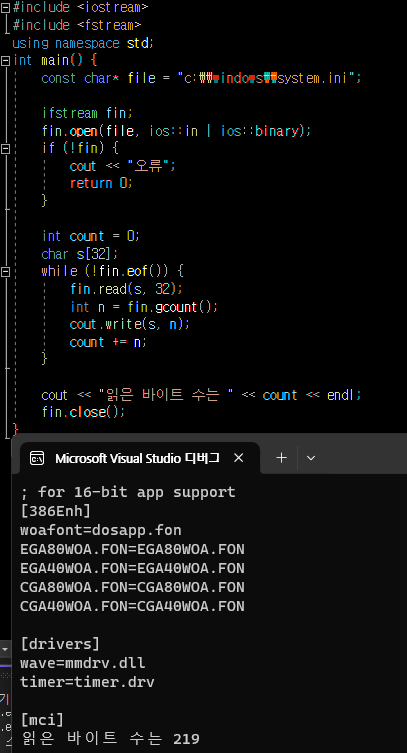
바이너리 I/O 방식은 데이터의 바이너리 값을 그대로 파일에 저장하거나, 파일의 바이너리 값을 그대로 읽어서 변수나 버퍼에 저장하는 방식이다.  
텍스트 파일이든 바이너리 파일이든 바이너리 I/O로 입출력 가능하다.

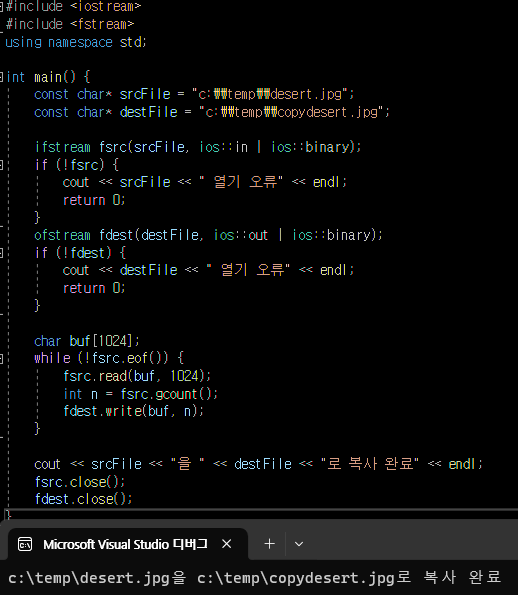
바이너리 I/O 모드를 열때에는 ios::binary를 사용하고 설정되지 않으면 디폴트가 텍스트 I/O이다.

예제 12-7)



get()/put()은 문자 혹은 바이트 단위로 파일 입출력한다.  
read()/write()는 블록 단위로 파일 입출력 한다.  
  
istream& read(char\* s, int n) – 최대 n개의 바이트를 배열 s에 읽어 들임  
ostream& write(char\* s, int n) – 배열 s에 있는 처음 n개의 바이트를 파일에 저장  
int gcount() – 최근에 파일에서 읽은 바이트 수 리턴

예제 12 -8)  


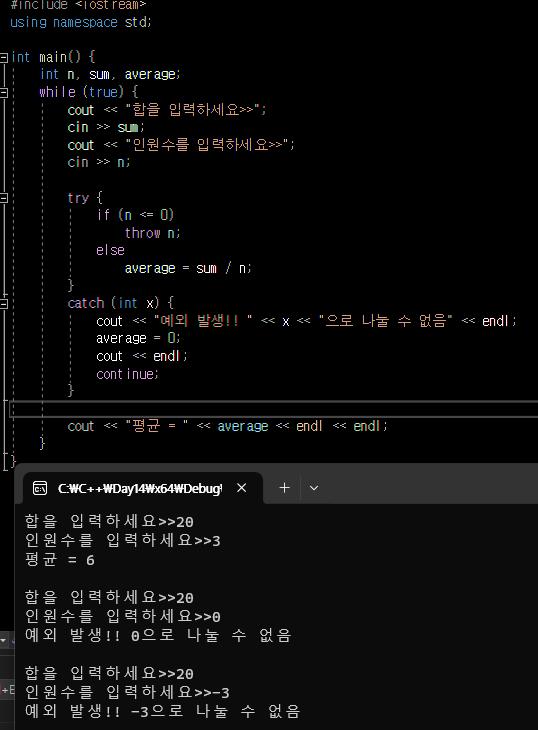
예제 12-9)  


<예외 처리와 C언어와의 링크 지정>

오류는 컴파일 오류와 실행 오류가 있다.  
컴파일 오류는 문법에 맞지 않는 구문으로 인한 오류이고  
실행 오류는 개발자의 논리 오류, 예외적으로 발생하는 입력이나 상황에 대한 대처가 없을 때 발생하는 오류, 실행 오류의 결과가 있다

실행 중, 프로그램 오동작이나 결과에 영향을 미치는 예상치 못한 상황 발생을 예외라 한다.

C++에서 예외 처리 기본 형식은  
try-throw-catch를 사용한다.  
try {} 블록은 예외가 발생할 가능성이 있는 코드를 묶고  
throw 문은 발견된 예외를 처리하기 위해, 예외 발생을 알리는 문장이고 try {} 블록 내에서 이루어져야 한다.  
catch() {} 블록은 throw에 의해 발생한 예외를 처리하는 코드이다.  
하나의 try{} 블록에 다수의 catch() {} 블록 연결이 가능하다.  
예외 처리를 함수나 클래스로 만들 수 있다. -> 장점은 프로그램의 작동을 명확히 하고 가독성을 높인다.

예제 13-4)  


extern “c”  
C 컴파일러로 컴파일할 것을 지시하는 코드 -> C 이름 규칙으로 목적 코드를 생성할 것을 지시함

사용법은

함수 하나만 선언  
extern “C” int f(int x, int y);

여러 함수들 선언  
extern “C” {

Int f(int x, int y);

Void g();

Char s(int []);

}

헤더파일 통째로 선언

Extern “C” {

#include “mycfunction.h”

}