
C++ 7-2

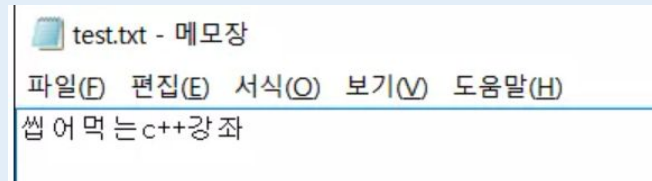
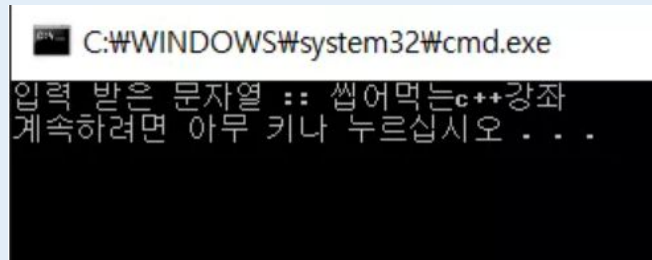
C++에서의 파일 입출력

fstream

```
// 파일에서의 입출력
#include <fstream>
#include <iostream>
#include <string>

int main() {
    // 파일 읽기 준비
    std::ifstream in("test.txt");
    std::string s;

    if (in.is_open()) {
        in >> s;
        std::cout << "입력 받은 문자열 :: " << s << std::endl;
    } else {
        std::cout << "파일을 찾을 수 없습니다!" << std::endl;
    }
    return 0;
}
```



fstream

```
#include <fstream>
#include <iostream>
#include <string>

int main() {
    // 파일 읽기 준비
    std::ifstream in("test.txt");
    std::string s;

    if (in.is_open()) {
        in >> s;
        std::cout << "입력 받은 문자열 :: " << s << std::endl;
    } else {
        std::cout << "파일을 찾을 수 없습니다!" << std::endl;
    }
}
```

```
in.close();
in.open("other.txt");

if (in.is_open()) {
    in >> s;
    std::cout << "입력 받은 문자열 :: " << s << std::endl;
} else {
    std::cout << "파일을 찾을 수 없습니다!" << std::endl;
}

return 0;
}
```

이진수로 읽기

```
#include <fstream>
#include <iostream>
#include <string>

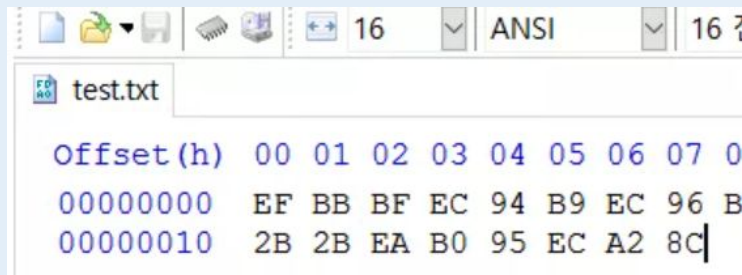
int main() {
    // 파일 읽기 준비
    std::ifstream in("test.txt", std::ios::binary);
    std::string s;

    int x;
    if (in.is_open()) {
        in.read((char*)&x, 4);
        std::cout << std::hex << x << std::endl;
    } else {
        std::cout << "파일을 찾을 수 없습니다!" << std::endl;
    }

    return 0;
}
```

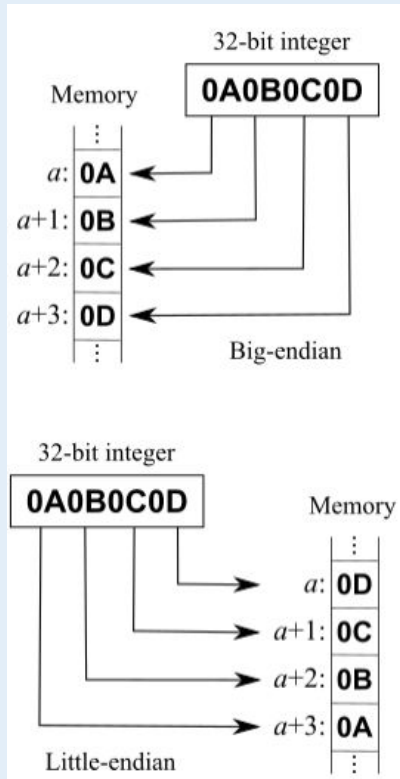
ecbfbbef

계속하려면 아무 키나 누르십시오 . . .



Offset(h)	00	01	02	03	04	05	06	07	08
00000000	EF	BB	BF	EC	94	B9	EC	96	B
00000010	2B	2B	EA	B0	95	EC	A2	8C	

이진수로 읽기



ecbfbbef

계속하려면 아무 키나 누르십시오 . . .

A screenshot of a hex editor window titled 'test.txt'. The interface shows a toolbar with icons for file operations and a status bar indicating a 16-bit offset and ANSI encoding. The main display area shows a table of memory offsets and their corresponding hexadecimal values.

Offset (h)	00	01	02	03	04	05	06	07	08
00000000	EF	BB	BF	EC	94	B9	EC	96	B
00000010	2B	2B	EA	B0	95	EC	A2	8C	

이진수로 읽기

```
#include <fstream>
#include <iostream>
#include <string>

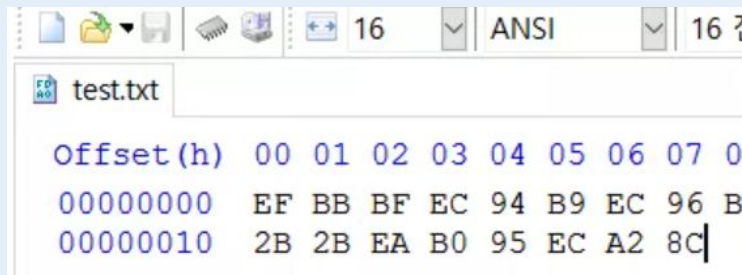
int main() {
    // 파일 읽기 준비
    std::ifstream in("test.txt", std::ios::binary);
    std::string s;

    int x;
    if (in.is_open()) {
        in.read((char*)&x, 4);
        std::cout << std::hex << x << std::endl;
    } else {
        std::cout << "파일을 찾을 수 없습니다!" << std::endl;
    }

    return 0;
}
```

ecbfbbef

계속하려면 아무 키나 누르십시오 . . .



Offset(h)	00	01	02	03	04	05	06	07	08
00000000	EF	BB	BF	EC	94	B9	EC	96	B
00000010	2B	2B	EA	B0	95	EC	A2	8C	

파일 전체 읽기

```
#include <fstream>
#include <iostream>
#include <string>

int main() {
    // 파일 읽기 준비
    std::ifstream in("test.txt");
    std::string s;

    if (in.is_open()) {
        // 위치 지정자를 파일 끝으로 옮긴다.
        in.seekg(0, std::ios::end);

        // 그리고 그 위치를 읽는다. (파일의 크기)
        int size = in.tellg();
```

```
        // 그 크기의 문자열을 할당한다.
        s.resize(size);

        // 위치 지정자를 다시 파일 맨 앞으로 옮긴다.
        in.seekg(0, std::ios::beg);

        // 파일 전체 내용을 읽어서 문자열에 저장한다.
        in.read(&s[0], size);
        std::cout << s << std::endl;
    } else {
        std::cout << "파일을 찾을 수 없습니다!" << std::endl;
    }

    return 0;
}
```

파일 전체 읽기

```
#include <fstream>
#include <iostream>
#include <string>

int main() {
    // 파일 읽기 준비
    std::ifstream in("test.txt");
    std::string s;

    if (in.is_open()) {
        // 위치 지정자를 파일 끝으로 옮긴다.
        in.seekg(0, std::ios::end);

        // 그리고 그 위치를 읽는다. (파일의 크기)
        int size = in.tellg();
```

C:\WINDOWS\system32\cmd.exe

아주
긴
내용의
글들이 이 파일에
있습니다~~
계속하려면 아무 키나 누르십시오 . . .

파일 전체 읽기

```
#include <fstream>
#include <iostream>
#include <string>

int main() {
    // 파일 읽기 준비
    std::ifstream in("test.txt");
    std::string s;

    if (in.is_open()) {
        // 위치 지정자를 파일 끝으로 옮긴다.
        in.seekg(0, std::ios::end);

        // 그리고 그 위치를 읽는다. (파일의 크기)
        int size = in.tellg();
```

```
        // 그 크기의 문자열을 할당한다.
        s.resize(size);

        // 위치 지정자를 다시 파일 맨 앞으로 옮긴다.
        in.seekg(0, std::ios::beg);

        // 파일 전체 내용을 읽어서 문자열에 저장한다.
        in.read(&s[0], size);
        std::cout << s << std::endl;
    } else {
        std::cout << "파일을 찾을 수 없습니다!" << std::endl;
    }

    return 0;
}
```

파일 전체 한 줄씩 읽기

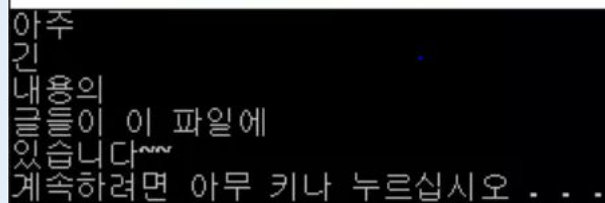
```
#include <fstream>
#include <iostream>
#include <string>

int main() {
    // 파일 읽기 준비
    std::ifstream in("test.txt");
    char buf[100];

    if (!in.is_open()) {
        std::cout << "파일을 찾을 수 없습니다!" << std::endl;
        return 0;
    }

    while (in) {
        in.getline(buf, 100);
        std::cout << buf << std::endl;
    }

    return 0;
}
```



아주
긴
내용의
글들이 이 파일에
있습니다~~~
계속하려면 아무 키나 누르십시오 . . .

파일 전체 한 줄씩 읽기

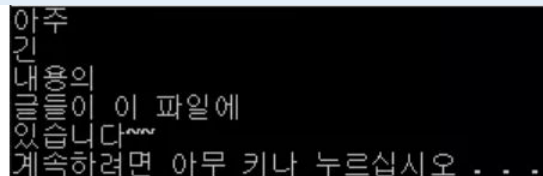
```
#include <fstream>
#include <iostream>
#include <string>

int main() {
    // 파일 읽기 준비
    std::ifstream in("test.txt");

    if (!in.is_open()) {
        std::cout << "파일을 찾을 수 없습니다!" << std::endl;
        return 0;
    }

    std::string s;
    while (in) {
        getline(in, s);
        std::cout << s << std::endl;
    }

    return 0;
}
```



아주
긴
내용의
내용이 이 파일에
있습니다~
계속하려면 아무 키나 누르십시오 . . .

파일에 쓰기

```
#include <iostream>
#include <fstream>
#include <string>

int main() {
    // 파일 쓰기 준비
    std::ofstream out("test.txt");

    std::string s;
    if (out.is_open()) {
        out << "이걸 쓰자~~";
    }

    return 0;
}
```



test.txt - 메모장

파일(F) 편집(E) 서식(O) 보기(V)

이걸 쓰자~~

파일에 쓰기

```
#include <fstream>
#include <iostream>
#include <string>

int main() {
    // 파일 쓰기준비
    std::ofstream out("test.txt", std::ios::app);

    std::string s;
    if (out.is_open()) {
        out << "뎃볼이기";
    }

    return 0;
}
```



test.txt - 메모장

파일(F) 편집(E) 서식(O) 보기(V) !

|이 걸 쓰자~~뎃볼이기