

3주차(2023-09-18)
실습 및 Pair Programming

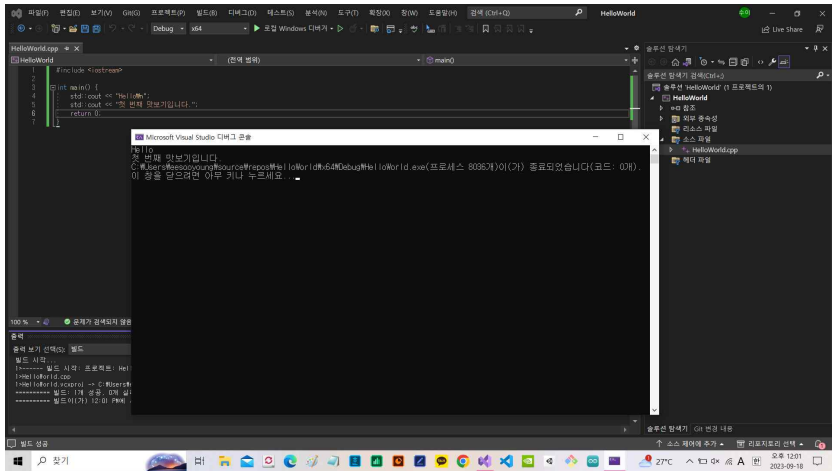
202084024 이수영

202104245 원준서

목차

- 예제 2-1(C++)
- 예제 2-2(C++)
- 예제 2-2(C)
- 예제 2-2(endl 제외)
- 예제 2-3(Pair Programming)
- iostream
- 예제 2-5(실패)
(Pair Programming)
- 예제 2-5(수정)
(Pair Programming)
- 연습문제 8번
(Pair Programming)

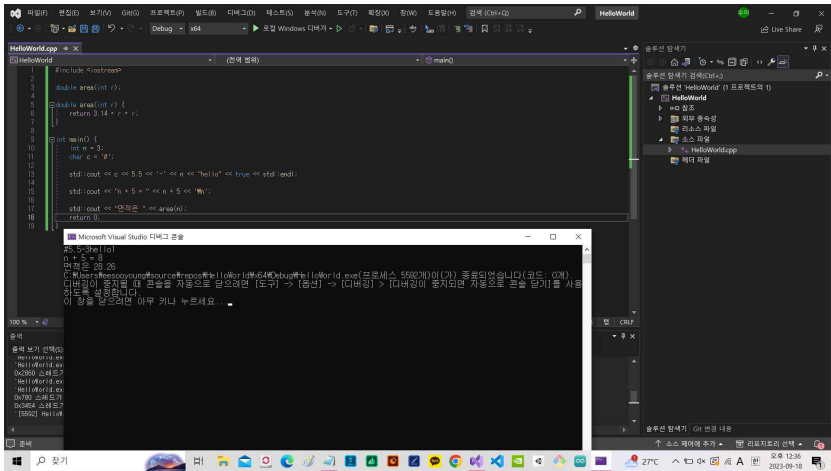
[1차시] 예제 2-1



[1차시] 예제 2-1

- C언어와 달리 C++에서는 `<iostream>`을 선언하며, `std::cout << “출력값”`의 형태가 `printf()`를 대신하고 있음을 새롭게 배웁니다.

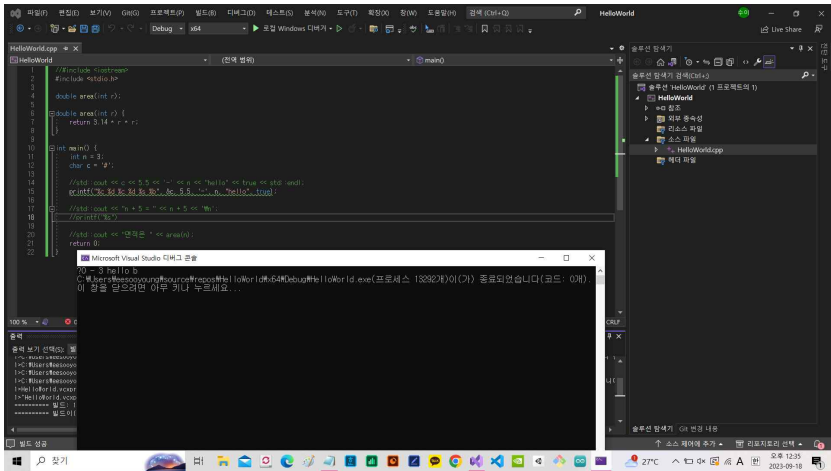
[1차시] 예제 2-2 (C++)



[1차시] 예제 2-2 (C++)

- << 를 한 행에 여러번 작성할 수 있음을 새롭게 배웠습니다.
- 특히 endl로도 줄 바꿈을 할 수 있다는 점이 흥미로웠습니다.

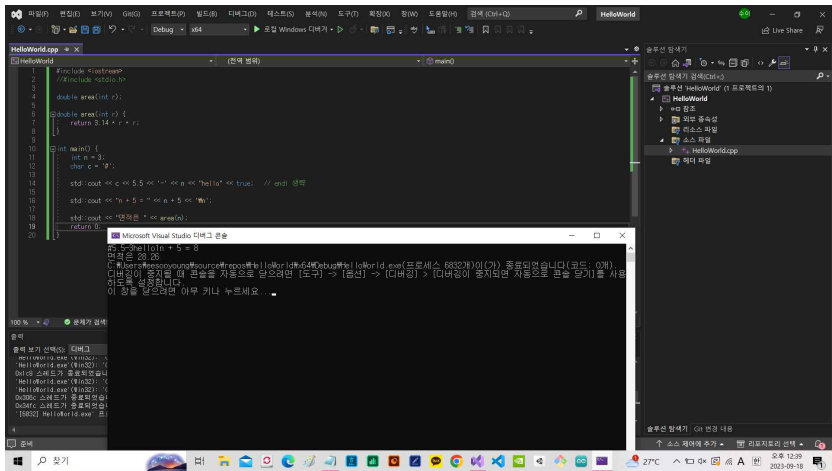
[1차시] 예제 2-2 (C)



[1차시] 예제 2-2 (C)

- C언어로 프로그래밍해보는 과정에서 데이터타입을 정확히 인지하고 있지 않아 컴파일 오류가 초반에 일어났습니다.
- C++을 공부할 때, 그 기반이 되는 C언어를 공부하고 있는데 C언어로 작성하는 코드를 C++로 어떻게 작성할 수 있을까? C++로 작성된 코드를 C언어로 어떻게 바꿀 수 있을까?
를 생각하면서 각 언어의 코드를 공부하다 보면 더 잘 기억할 수 있을 것 같다는 점을 느꼈습니다.
printf(), scanf()가 cout, cin 객체에 대응되는 점과 같습니다.

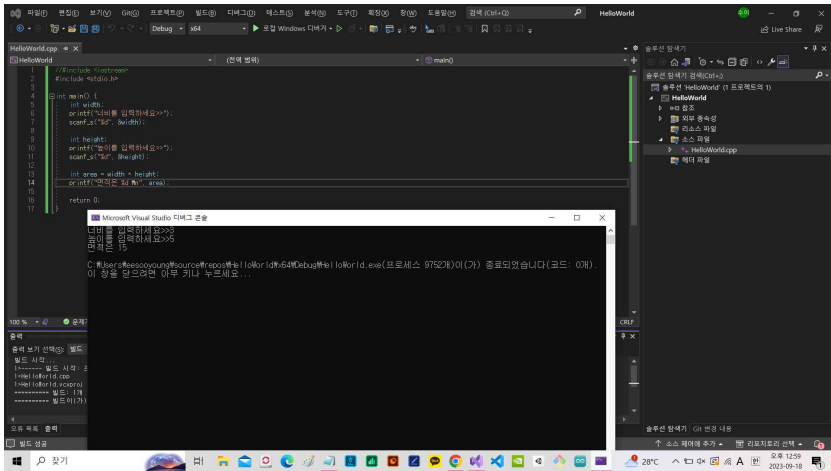
[1차시] 예제 2-2 (endl 제외)



[1차시] 예제 2-2 (endl 제외)

- endl을 사용하면 실행결과에 개행(줄 바꿈)이 일어나지만, 사용하지 않으면 줄 바꿈이 일어나지 않음을 배웠습니다.
- 앞선 실습에서 느꼈던 것처럼
C++ 문법을 이해할 때 다른 언어와 비교할 때
C++ 문법을 더 잘 이해할 수 있는 것 같다고 생각했습니다.
C++의 endl을 사용했을 때와 사용하지 않았을 때는
Java의 println(), print()와 유사하다는 점을 연상한다면
endl을 더 오래 기억할 수 있을 것 같습니다.

[1차시] 예제 2-3 (Pair Programming)



[1차시] 예제 2-3 (Pair Programming)

- 함께 Pair Programming을 하는 과정을 통해서 서로 C언어의 개념을 보완해가며 코드를 작성하는 기쁨을 느꼈습니다. C++ 코드만 공부하다가 예전에 배운 C언어 코드로 바꾸는 경험이 많지 않다보니 초반에는 낯설어서 막막했습니다. scanf()에서 정수는 %d(decimal), &입력받은 변수명 형태로 작성하는 과정을 함께 하니 혼자 작성해야 하는 상황보다 부담감이 덜어진 것 같았습니다. 그리고 변수 width, height를 먼저 선언하기 보다는 관련된 코드 printf(), scanf() 바로 앞에 각각 작성하면 코드를 이해하는 데도 훨씬 수월하다는 점을 느꼈습니다.

[2차시] iostream 공부

Browser tabs: sylee2020, cplusplus.com/reference/ios/

Address bar: cplusplus.com/reference/ios/

Navigation icons: Back, Forward, Reload, Home, Search, etc.

Search engines: B, A, H, J, 포름, Y, Daum, NAVER, Google, Gmail, YouTube, pc, cplusplus.com, etc.

Left sidebar menu:

- Tutorials
- Reference
- Articles
- Forum

Reference section:

- C library:
- Containers:
- Input/Output:
 - <fstream>
 - <iomanip>
 - <ios>
 - <iosfwd>
 - <iostream>
 - <ostream>
 - <sstream>
 - <streambuf>
- Multi-threading:
- Other:

<ios> section:

- types
 - basic_ios
 - fpos
 - io_errc
 - ios
 - ios_base
 - streamoff
 - streampos
 - streamsize

Header: <ios>

Input-Output base classes

Header providing base classes and types for the IOSTream hierarchy of classes:

```
graph TD
    ios["<ios>"] --> ios_base["ios_base"]
    ios --> ios["ios"]
    ios_base --> istream["<istream>"]
    ios_base --> ostream["<ostream>"]
    ios_base --> streambuf["<streambuf>"]
    istream --> ifstream["<ifstream>"]
    istream --> stringstream["<stringstream>"]
    ostream --> ofstream["<ofstream>"]
    ostream --> stringstream["<stringstream>"]
    streambuf --> stringbuf["<stringbuf>"]
    cin["cin"] --> istream
    cout["cout, cerr, clog"] --> ostream
```

Types

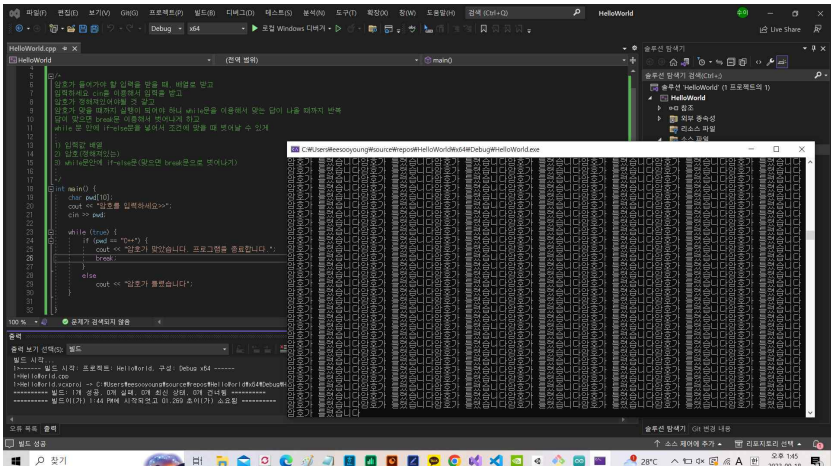
Footer: CITEO 광고, 광고 선택하기, Ad choices

System tray: Windows logo, Search, Taskbar icons, Date/Time: 2023-09-18, 12:22

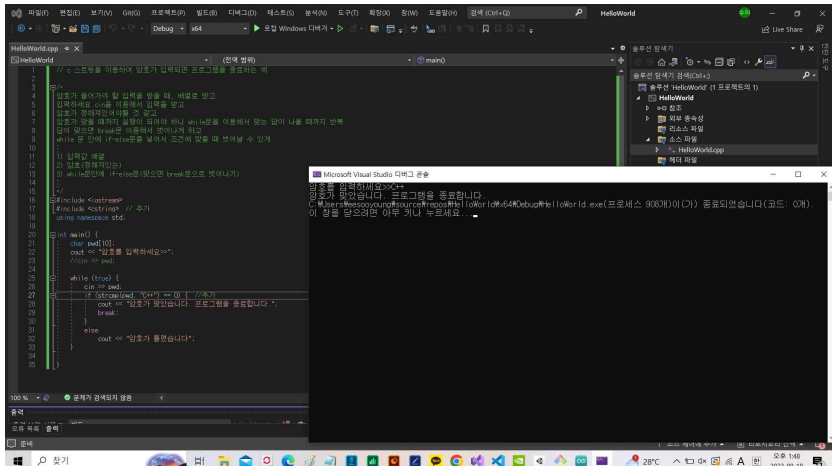
[2차시] iostream 공부

- C library를 확인할 수 있는 사이트인 cplusplus를
실습 중에 확인하는 과정에서
코드를 작성하는 가이드를 새롭게 알게 돼서 좋았습니다.
<iostream> 역시 <ios>에 속해있다는 점을
계층화된 그림으로 확인하면서
C++ 언어 문법을 활자로 볼 때보다 더 쉽게 이해했습니다.

[2차시] 예제 2-5 (실패)



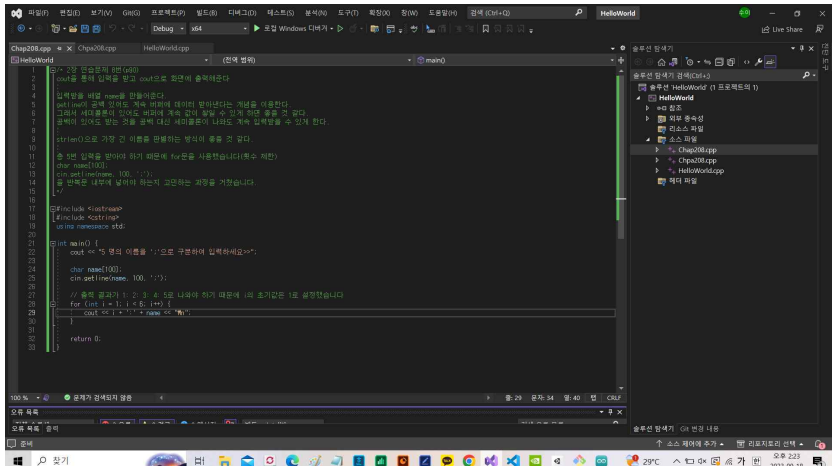
[2차시] 예제 2-5 (수정)



[2차시] 예제 2-5 (실패)(수정)

- Pair Programming 과정에서 서로 어떤 점을 코드로 구현해야 하는지 서로 나눈 내용은 주석으로 작성했습니다.
- 21행 `cin>>pwd;`를 26행으로 옮겼습니다. 초반에는 '암호를 입력하세요'의 입력값을 `pwd`에 바로 담는다는 의미로 21행에 작성했지만, 조건문에 해당하는 변수라는 점에서 조건문 내부로 옮겼습니다.
- 그리고 `strcmp()`를 처음에 떠올리지 못했지만, 입력값이 이미 정해진 패스워드 문자열과 일치하는지 여부를 판단할 때 `strcmp()`를 사용해야 한다는 점을 뒤늦게 깨달았습니다. 그런데 오히려 원한 결과를 바로 얻지 못하고 시행착오를 겪었지만 그랬기에 `strcmp()`를 오래 기억할 수 있었습니다.

[3차시] p90 2장 연습문제 8번



[3차시] p90 2장 연습문제 8번

- Pair Programming 과정에서 서로 어떤 점을 코드로 구현해야 하는지 서로 나눈 내용은 주석으로 작성했습니다.
- cin과 cin.getline()의 차이를 배운 후에 한 실습이었기 때문에 cin.getline()이 반드시 '공백'이 있어도 계속 입력을 받는다는 것 외에도 ';' (세미콜론)이 있어도 계속 입력을 받는다는 걸 새롭게 배웠습니다.
- 5번 입력받겠다고 해서 for문을 사용했는데, 1: 2: 3: 4: 5:를 어떻게 구현하면 좋을지 계속 고민하는 과정을 거쳤습니다.
- 가장 긴 이름을 찾을 때는 strlen()을 구현하고 싶었는데 어떤 식으로 구현할 지 구체적인 코드 작성이 막혔습니다.

느낀 점

- 수업시간에 배운 내용이 예제를 실습하는 과정에서 곧바로 적용되는 점을 보면 그 개념을 확실히 익힐 수 있었습니다.
- 반면, 응용해야 하는 문제의 경우에 막히는 부분이 있을 때 `strcmp()`를 어떻게 구현하면 좋을지 2-3차시 강의에서 서로 논의하고 알아보는 과정에서 더 깊이 공부할 수 있었습니다.
- 서로 아는 점을 함께 나누고 모르는 점을 보완해나가면서도 모두 아직 모르는 부분은 관련된 내용을 추가로 알아가보고 공부하는 과정이 필요하다는 점을 느꼈습니다.
그리고 그 과정에서 익힌 내용은 여러 시행착오를 함께 느꼈기 때문에 확실히 기억할 수 있다는 점을 느꼈습니다.