



C++ 프로그래밍

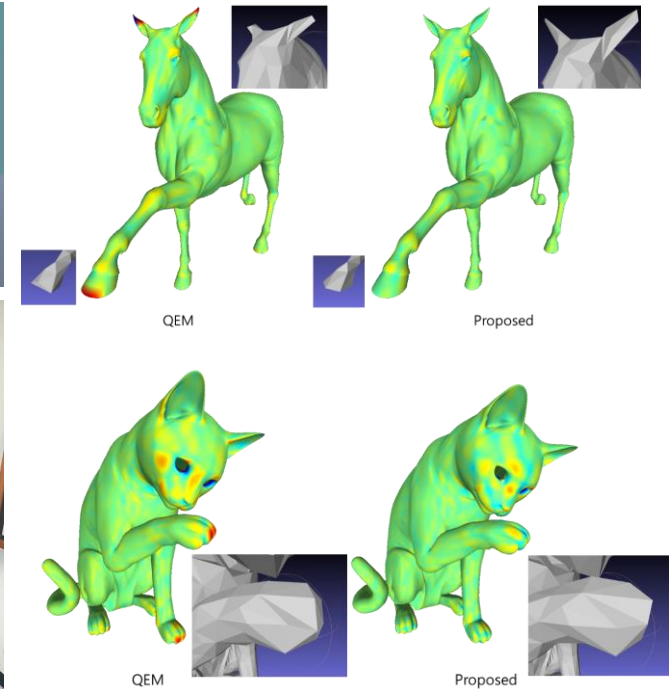
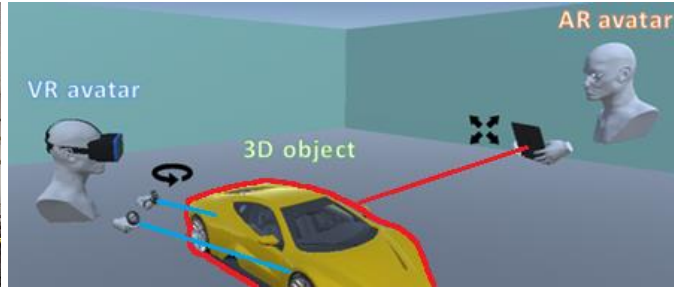
김 형 기

hk.kim@jbnu.ac.kr

Welcome!

- 담당교수 : 김형기

- 연구 분야 : 가상현실, 컴퓨터 그래픽스, 3D 딥러닝
- 공학 7호관 625호, 063-270-3385
- hk.kim@jbnu.ac.kr



학부 연구생 및 대학원생 모집

- VCLab에서 학부 연구생 및 대학원생을 모집합니다!
 - <https://sites.google.com/view/vclab-jbnu>
 - 과제 연구 및 업무보조 + 개인 프로젝트
 - 학술활동/인건비/교육 지원
 - 모집 분야
 - VR/AR 비대칭 협업 환경 구축 기술 연구
 - 360도 카메라 기반의 Teleoperation이 가능한 원격 협업 환경 구축 기술 연구
 - 게임 엔진(Unity3D), 임베디드 프로그래밍 교육 지원
 - AR 기반 철도 유지보수 지원 시스템 구축 기술 연구
 - 공간 컴퓨팅 기반의 유지보수 지원 시스템 구축 기술 연구
 - 게임 엔진, MS 홀로렌즈 관련 교육 지원
 - 3D 형상 생성 기술 연구
 - 인공지능 기술을 활용한 3D 모델 생성 기술 연구
 - 3D 딥러닝 관련 교육 지원

Course Introduction

Course introduction

- C++ 프로그래밍

- C 언어 기초 이수 학생을 대상으로 함
- **메모리 레이아웃과 명령어에 따른 메모리의 변화 이해 ← 초반부 주요 목표**
- **객체지향 프로그래밍 기법의 이해 ← 중/후반부 주요 목표**
- 실습을 통한 구체적인 이해
 - Modern C++을 기준으로 한 기능 이해

- 이 강의의 중요성

- 자료구조 / JAVA프로그래밍 / 윈도우즈프로그래밍 / 알고리즘 / 컴퓨터그래픽스 등 강의의 기반
- 제대로 이해하면 학교 생활이 편하고 재미있어 집니다.

Course introduction

- 프로그래밍이 어려우신가요?

- (1학년 수준에서) 코딩이 어렵다고 하시는 분들을 분석해 본 결과...
- 가장 많이 보이는 유형은 아래와 같았습니다.

- 들을때는 이해가 되는 것 같은데 직접 해보려고 하면 안돼요 ㅜㅜ
- 예제랑 비슷하게 코드를 작성했는데 제가 생각한 것과 다르게 동작해요 ㅜㅜ

- 원인은?


- 명령문이 어떤 일을 하는지 정확히 모르고 외워서, 또는 느낌적으로(?)만 이해했기 때문
- 즉, 실제로 (메모리에서) 발생하는 일을 정확히 모르는 상태에서 모호하게 사용하기 때문!



Course introduction

- 강의 초반부의 목표

- 메모리 레이아웃과 명령어에 의한 메모리 변화 이해 + 디버거를 사용한 관찰 방법을 배운다!
- 질문) 아래 3번 라인의 명령문을 실행하면 어떻게 될까요?



```
1 int main()
2 {
3     int a = 10;
4 }
```


답변) a라는 정수형 변수를
10으로 초기화하여 생성한다.

현재

Course introduction

- 강의 초반부의 목표

- 메모리 레이아웃과 명령어에 의한 메모리 변화 이해 + 디버거를 사용한 관찰 방법을 배운다!
- 질문) 아래 3번 라인의 명령문을 실행하면 어떻게 될까요?

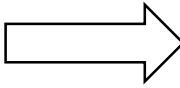


```
1 int main()
2 {
3     int a = 10;
4 }
```

답변) a라는 정수형 변수를
10으로 초기화하여 생성한다.



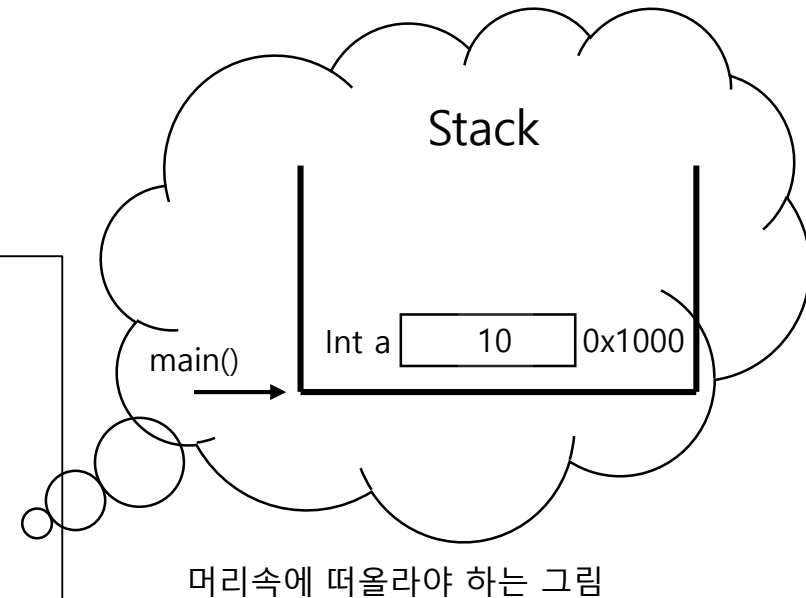
현재



답변) 스택 메모리의 어떤
주소(ex, 0x1000)에 4
바이트 메모리가 할당되고
10이라는 값이 저장된다.
해당 주소에 a라는 이름이
붙는다

**"스택 메모리", "주소", "바이트"
에 겁먹을 필요가 전혀 없습니다.*

4주차까지 강의를 들은 후



Keep in mind

- 여러분이 기억해야 할 세가지

- 강의를 “구경”하지 마세요!

- 피아노를 치는 법을 본다고 피아노를 잘 치게 되는 것이 아닙니다. 본인이 쳐보아야 합니다.
 - 제가 타이핑하는 내용을 따라서 모두 타이핑 하고, 수정하고, 실행해 보세요.

- 스스로를 너무 믿지 마세요!

- 제가 강의에서 “직접 해보세요/따라 해보세요” 라고 말하는 내용은 꼭 해보세요.
 - “나는 다 이해했으니 안해봐도 됨ㅋ”, “딱 보니까 이렇게 되겠네”하고 넘어가면 안됩니다.

- 질문하세요!

- 익명도 좋습니다. 이해가 안되는 내용은 바로바로 질문하세요.
 - 강의는 100% 완벽할 수 없습니다. 저는 여러분의 머리속을 볼 수 없습니다.
 - 여러분의 질문과 그에 대한 답변까지 완료되어야 강의가 100% 완성됩니다.

Keep in mind

● Long way to go

1. 기술의 빠른 변화

- 기본 개념 이해가 없다면 새로운 내용을 배울 때 처음부터 다시 시작해야 함

2. 학점이나 실력이나: 장기적으로 필요한 것은 실력임

- 편안한 강의를 원하는 분은 다른 분반으로...

- 학점은 비율에 맞추어 부여하나, 쉽게 높은 점수를 얻는 것은 제 강의의 목적과는 다릅니다.

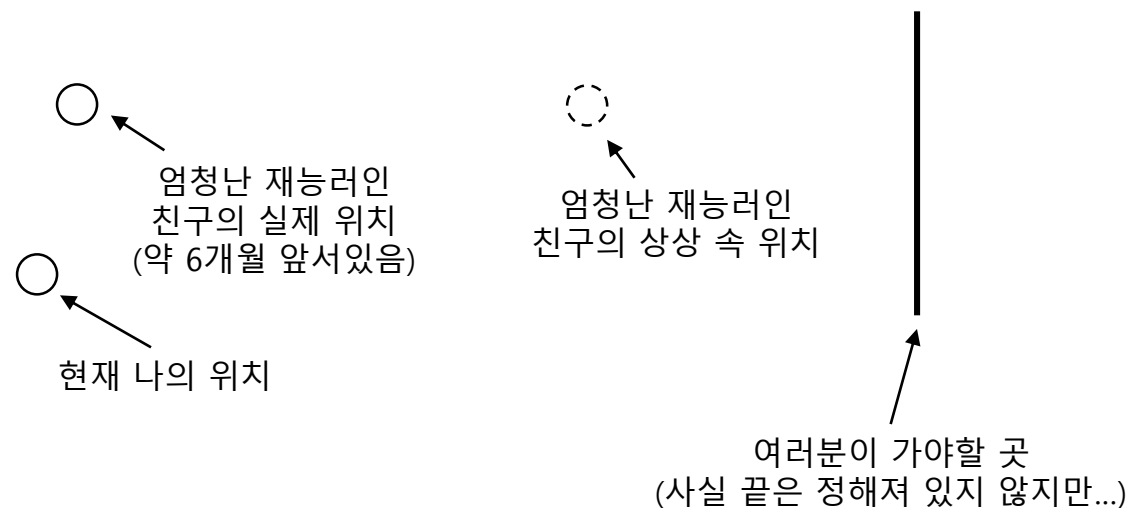
3. 재능을 탄다? 대부분은 오해입니다.

- 옆의 친구가 엄청난 재능러 같나요?

- 갈 길과 비교하면 미미한 차이

- 강의에서 배울 내용을 그 친구는 이미 알고있다?

- 길어야 고작 한 학기 차이일 뿐



Why do we learn C++?

- C++ 사용 분야
 - 아마존, 인텔, 구글, IBM, MS, Apple...
 - 윈도우, 리눅스, 포토샵, DB, 게임 엔진...
 - 컴퓨터 그래픽스, 머신 러닝, ...
- 빠르고, 유연하며 높은 확장성 보유
- 절차적 & 객체지향 프로그래밍
- 복잡하다 → 어렵다 → Blue ocean (상대적으로...)

C++ History

- C
 - Ken Thompson and Dennis Ritchie, 1972
 - 빠르게 동작하고, 간단하며 플랫폼 독립적인 코드 작성을 위해 탄생
 - 단점
 - 대규모 프로그램의 작성 및 유지보수의 어려움
- C++
 - Bjarne Stroustrup, 1983
 - 빠르게 동작하고, 간단하며 플랫폼 독립적인 코드 작성 여전히 가능
 - + 고수준의 프로그래밍 기능 (클래스, 다형성, 제네릭...)
 - 단점
 - 직접 경험해 봅시다!



Ken Thompson and Dennis Ritchie



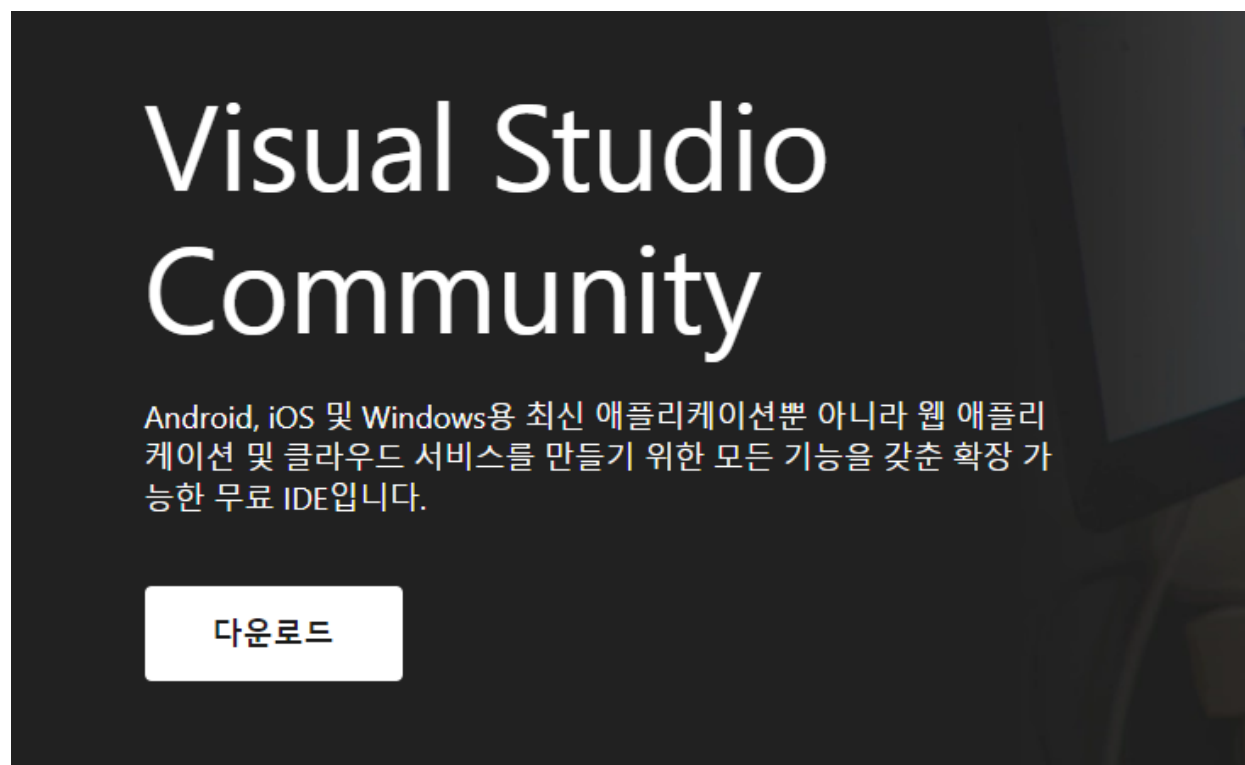
Bjarne Stroustrup

IDE – Visual Studio Community 2022

- 실습에 사용할 기준 IDE
 - 이전 버전의 Visual Studio를 사용해도 무방합니다.
 - <https://visualstudio.microsoft.com/vs/community/>

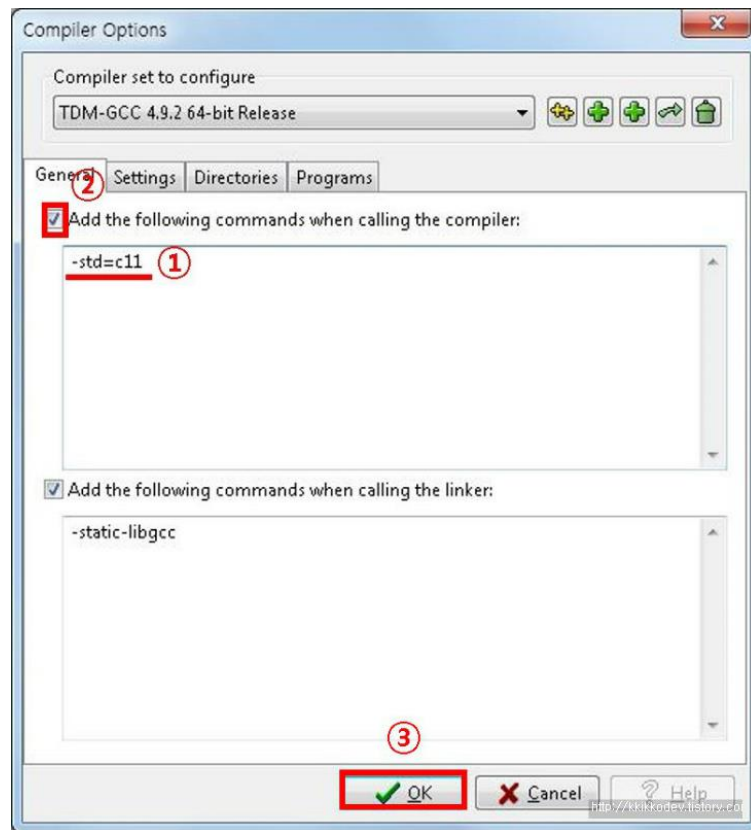
**DevC, VSCode 사용하면 안되나요?*

- 정 본인이 원하시면 사용해도 되지만, 디버깅 툴을 사용해 자주 설명드릴 것이므로 추천하지 않습니다.
- 변화에 적응을 하는 것도 능력
- DevC는 실무에서 사용 안함



DevC 사용하는 경우 설정

- Tools→Compiler Options 메뉴 클릭
- 아래 사진과 같이 `-std=c11` 추가한 뒤, OK 클릭
 - 다른 IDE(ex, VSCode) 에서 GCC / G++ 컴파일러 사용하는 경우에도 마찬가지로 설정 바람



Syllabus & Grading

Syllabus – up to midterm

- 1주 : 강의 소개, C++ 프로그래밍 개요와 변수
- 2주 : 디버깅과 C++ 기본 문법
- 3주 : 함수
- 4주 : 포인터와 참조자
- 5주 : 객체지향 프로그래밍 개요, 클래스와 객체 Part 1
- 6주 : 클래스와 객체 Part 2
- 7주 : 상속 Part 1
- 8주 : 중간평가

Syllabus – up to final

- 9주 : 상속 Part 2
- 10주: 다형성
- 11주 : 연산자 오버로딩 Part 1
- 12주 : 연산자 오버로딩 Part 2
- 13주 : 템플릿과 Generic Programming
- 14주 : 표준 템플릿 라이브러리 (STL)
- 15주 : 기말평가

How to ask a question?

- 나쁜 질문

- "비주얼 스튜디오 설치 하는데 오류가 나요 ㅜㅜ"
- "상속이 잘 이해가 안돼요..."

- 좋은 질문

- 답변자가 추가적인 질문을 하지 않도록 최대한 상세히(스크린샷 등)
- 문제를 해결하기 위해 시도해 본 방법들은 모두 설명
- **중요!!** 구글링 해보기 전에는 절대 질문하지 말 것!

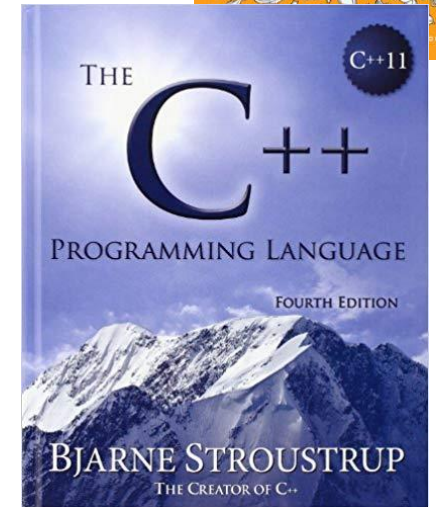
- 좋은 질문 하는 방법

- 문서를 못쓰면 주니어 (<https://www.youtube.com/watch?v=2yxUkrCvNa8>)
- <https://ppt21.com/pb/pb.php?id=freedom&no=74476&page=29>

Materials & Resources

Textbook & Resources

- 윤성우의 열혈 C++ 프로그래밍 (구매 추천)
 - 강의에서 다루는 내용 수준과 유사
 - 강의 순서는 약간의 차이가 존재(이해에 문제 X)
- The C++ Programming Language
 - 번역본 있음
 - Advanced
- <https://boycoding.tistory.com/>
- <https://modoocode.com/>

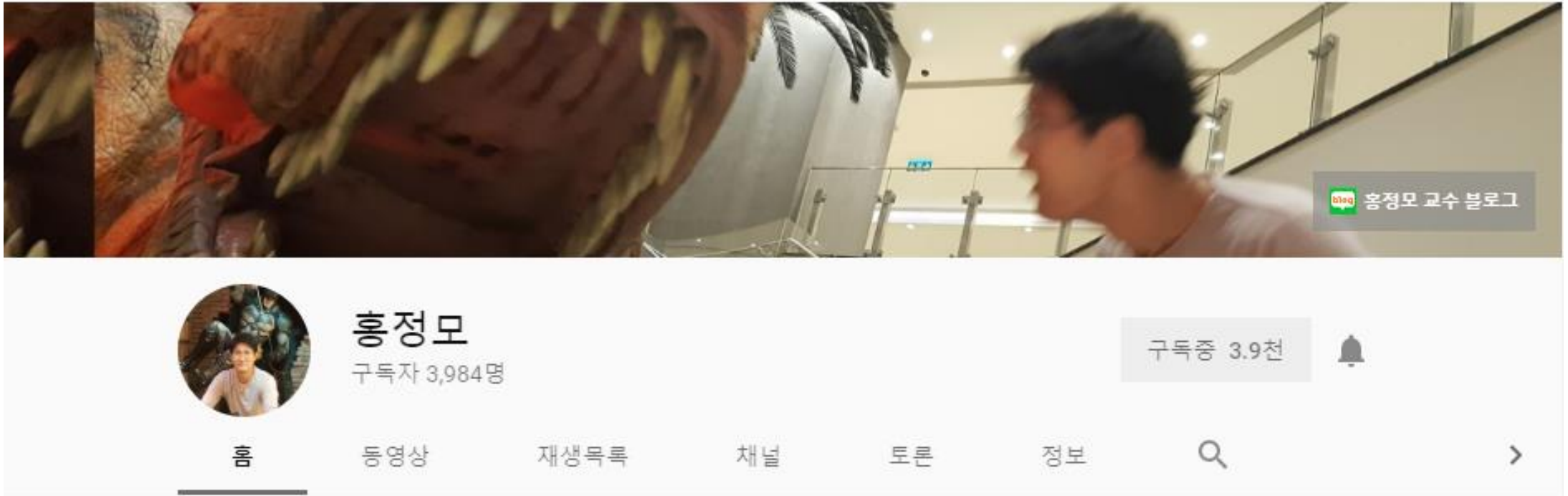


Textbook & Resources

- 홍정모 교수님 유튜브 채널

- 따라하며 배우는 C++

- https://www.youtube.com/channel/UCg6llhycdYiK_nWB3spjlqA/playlists



Textbook & Resources

- The Cherno 유튜브 채널

- C++ series

- <https://www.youtube.com/watch?v=18c3MTX0PK0&list=PLlrATfBNZ98dudnM48yfGUldqGD0S4FFb>



The Cherno ✓

구독자 28.5만명

구독중



Textbook & Resources

- 제 이전 강의 영상

- <https://youtube.com/playlist?list=PLMcUoebWMS1nzhIx-NbD4KBGEP1UCUDF>



C++ 프로그래밍

5. OOP - 1

(6) 생성자 오버로딩

전북대학교 컴퓨터공학부
김 형 기

Recommendation

- 강의 중

- 강의에서는 이해에 집중! 중간중간 꼭 따라서 타이핑 해볼 것!
- 복습 시 필기 없이 이해가 어려울 것 같은 부분만 필기
- 코드를 집중해서 이해

- 복습

- 각 섹션 끝
- 강의 끝
- 다음 강의 하루 전
- 다음 강의 시작 전
- 8번 복습 → 90% 기억!