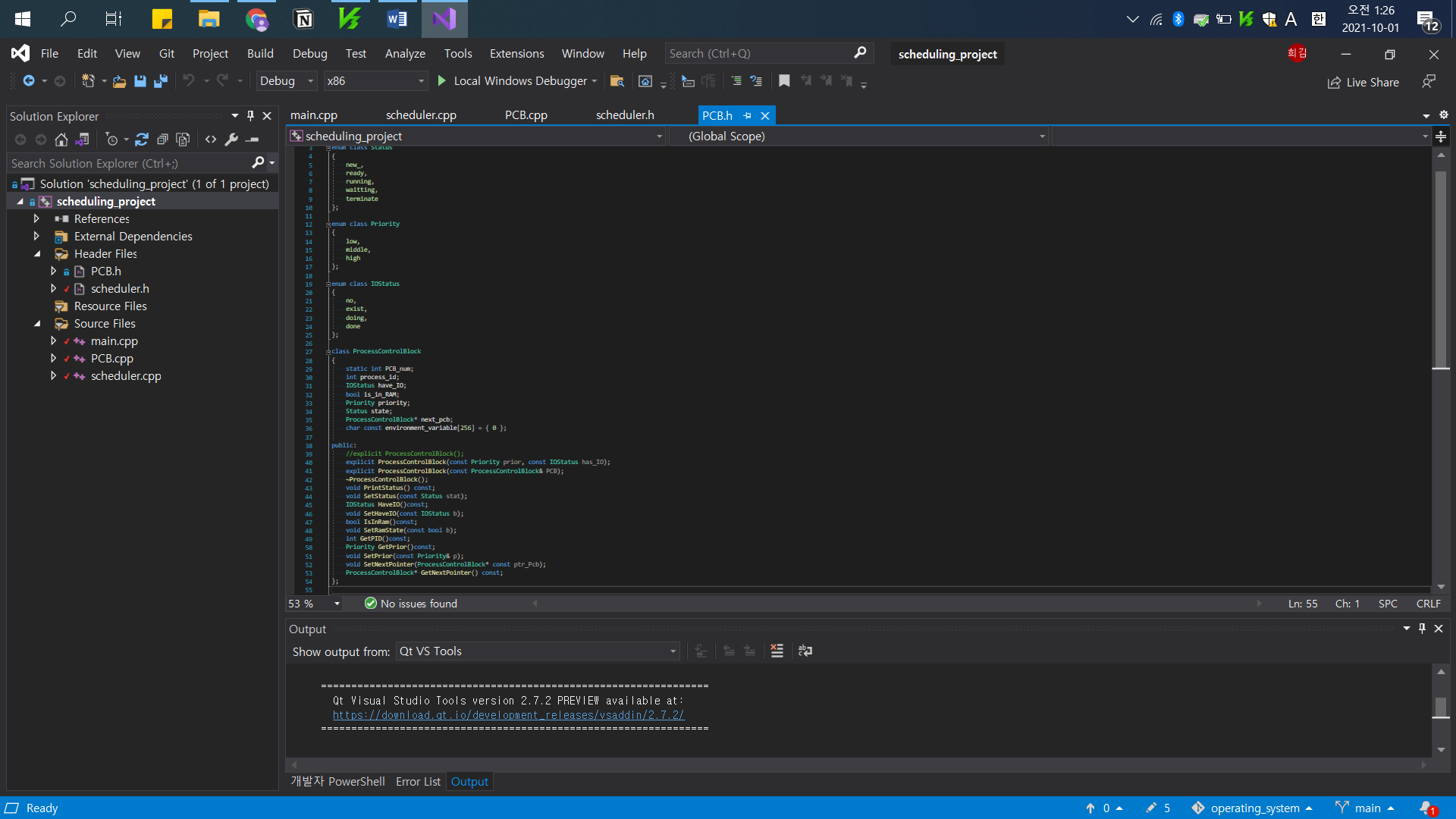
2020105695\_김희성

Scheduling Queues

# PCB Implementation

Code: PCB.h, PCB.cpp



차별점:

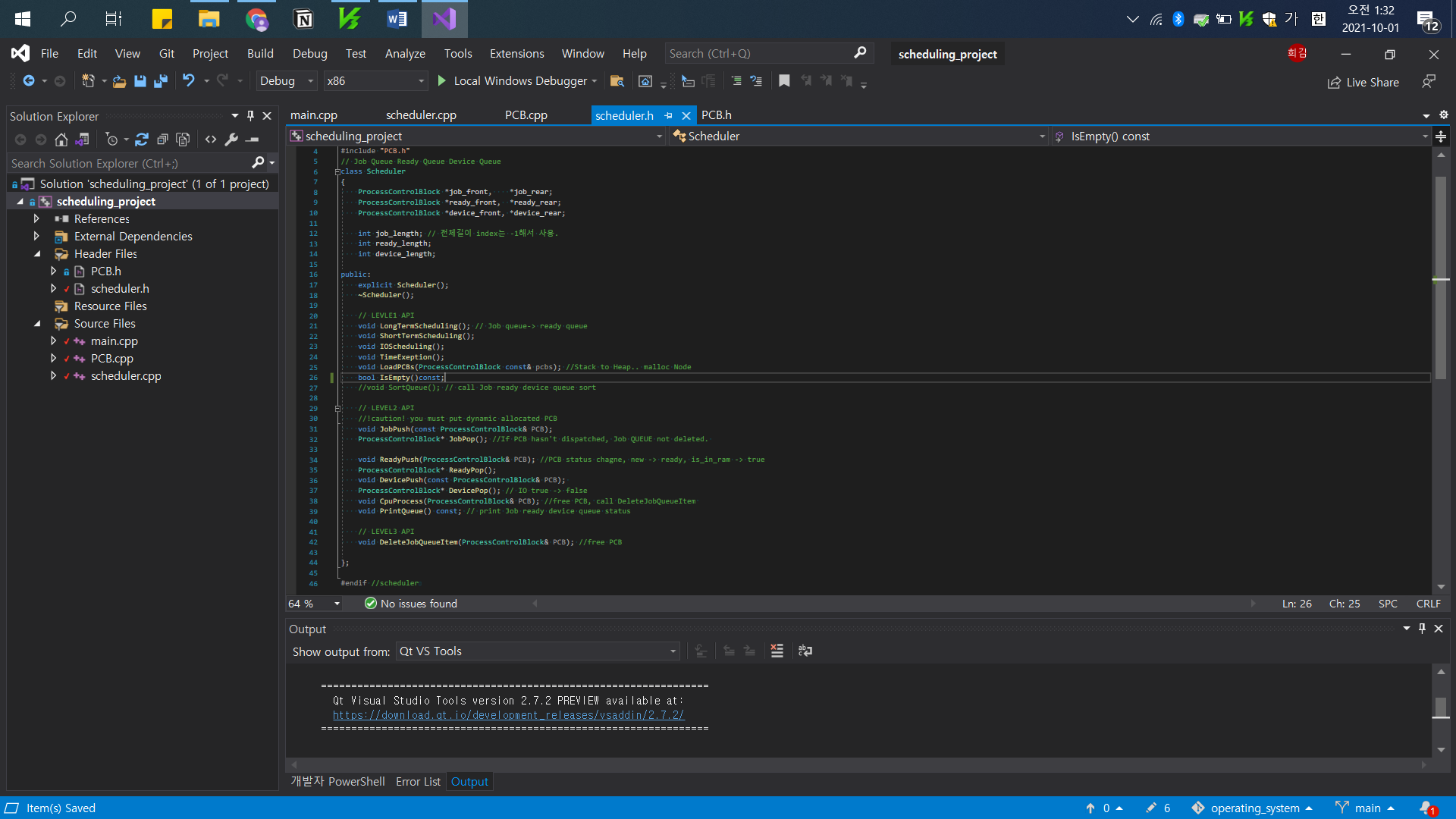
* window에선 환경 변수가 부모 프로세스에서 자식 프로세스로 상속된다고 나와있었다. 그래서 환경 변수에 관한 term도 넣었다. 총 프로세스의 개수와 process id 등등 많은 정보를 담았다.
* Priority 정보와 IO 정보가 scheduling 도중 바뀐다. Ex) ready queue에서는 ready 상태, IO 상태에선 waiting 상태로 바뀐다.

보완점:

* Context switching 과정에서 register(SP, PC) 등의 실행 중인 process가 기록된다고 배웠다. 하지만 이것 까진 구현하지 못했다.
* Job scheduler가 secondary storage에 저장돼야 하지만 RAM에 저장된다.

# Scheduling Queues Implementation

Code: scheduler.h, scheduler.cpp



차별점:

* Linked list 기반으로 각각의 queue를 만들었다. 보통의 linked list로 구현된 queue는 push할 때 동적 할당되고 pop할 때 해제된다. 이동 과정에서 동적 할당을 여러 번 하면 overhead가 클 것 같아서 job Queue에 올릴 때와, Ready Queue에 올릴 때만 동적 할당을 진행했다. (자료구조 같이 수강하고 있어서 확실 한진 모르겠습니다!)

보완점:

* Scheduling하는 과정에서 priority로 round robin하는 과정을 구하지 못했다.
* Priority 기반 scheduling 기능을 추가하고 싶었다. 하지만 sorting 알고리즘을 아직 배우는 중이다.

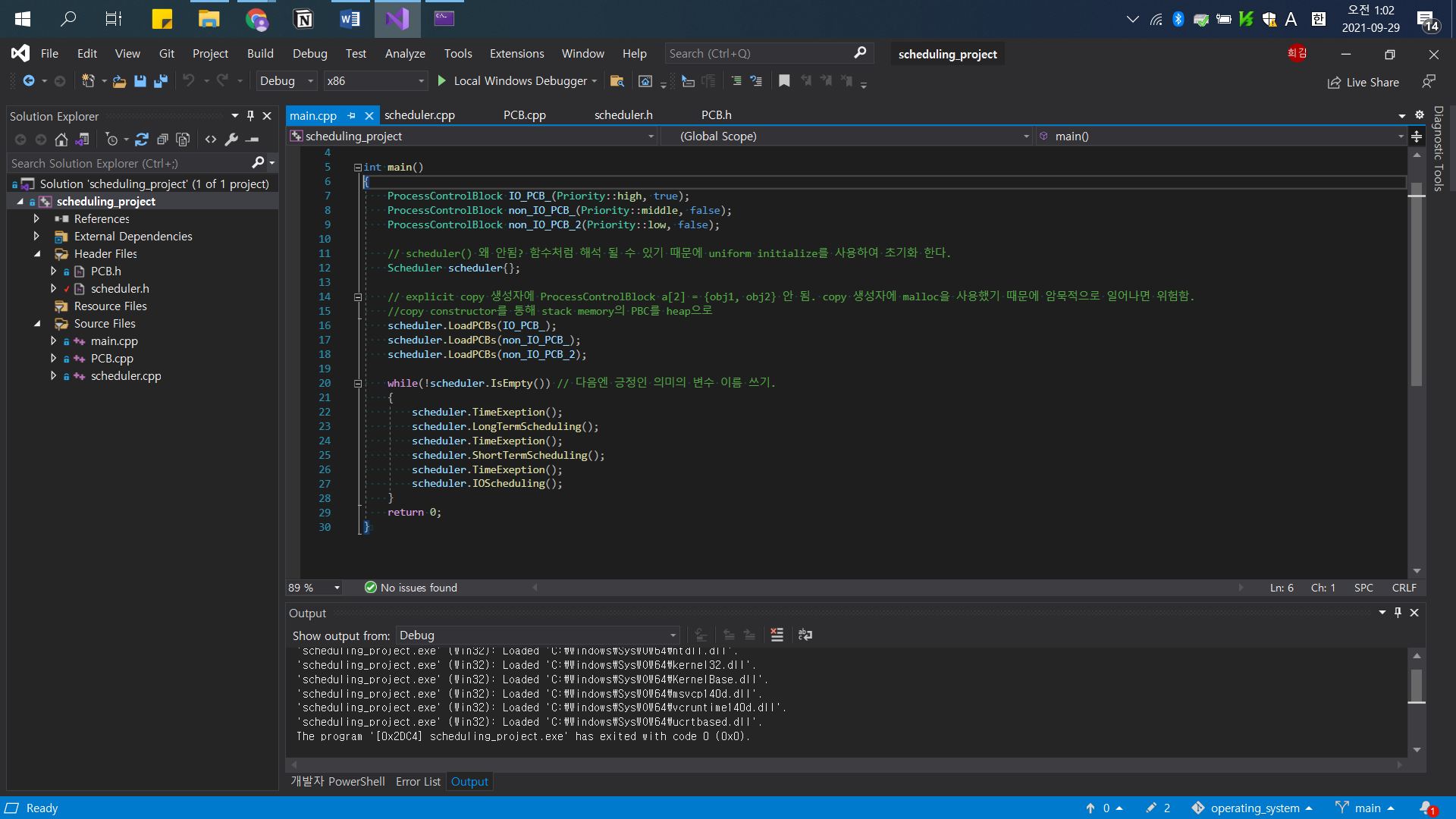
배운점:

* Container가 node Pointer를 가질 때, constructor에 new연산을 통해 넣어 놓는 것이 좋을 것 같다.
* 복사 생성자는 Stack에서 Heap으로 할당해줄 수 있다. Ram에 load 가능하다.

# Report

PBC가 한 개씩 진행되는 것을 Exeption을 통해 나타내봤다.

Code: main.cpp



결과) ->result.txt, scheduling\_project.exe

