

## Project #1 Multi-Thread Programming

- 과제 개요

- 멀티쓰레드 기반의 프로그래밍
- 기존 개발된 프로그래밍 소스코드를 참조하여, 기능을 추가 함
- 개발언어(C/C++/C#/Java/Android/Python 등)
- 제출형태:
  - :SOFT COPY (실행가능한 이클립스 또는 비주얼스튜디오 프로젝트 파일 그대로 제출) &
  - :HARD COPY(모두 제출), HWP or DOCX
- 제출기한: 05.13.금.18시 (e-campus: 05/13/금/18:00). 307호 앞 레포트 수거함

- 보고서 작성 지침

(아래 내용을 참고하여 목차를 자유롭게 구성할 것)

### 1. 개요:

- 프로그램의 개요를 설명
- 개요 그림과 주요 기능에 대한 내용을 간략하게 서술함

### 2. 본문: 기존 프로그램 또는 본인이 직접 개발한 프로그램의 주요기능 설명(소스코드 일부 포함)

- 직접 개발한 경우 주요 기능의 모두를 주요 소스코드 일부와 함께 설명
- 기존 프로그램의 소스코드를 사용한 경우, 기존 개발된 주요기능에 추가 기능을 개발하기 위해 본인이 작업한 내용을 위주로 설명을 추가할 것

### 3. 실험

- 주요 기능에 해당하는 실험결과(스크린 샷)를 나누어 설명

### 3. 결론

- 상기 내용을 간단하게 요약 정리하고, 쓰레드 기반 프로그래밍의 개발 경험에 대한 개인적 소견을 서술함

- 구현 요구사항

- 쓰레드 생성: 게임에 적합한 수 만큼의 쓰레드 생성
- Critical Section 처리
  - : 동시에 2개 이상의 Thread가 접근하는 경우에 대한 처리 기능
  - : 인위적으로 critical section에 동시에 접근할 수 있는 상황을 만들어서 테스트 함

- 평가 (비율)

- 구현 로직: Thread 생성 및 진행과정 (40%)
- GUI 구성 (30%)
- 보고서 (30%)

※ 쓰레드 기반 구현 가능 프로그램

프로그램	설명	참고
채팅 프로그램	<ul style="list-style-type: none"> <li>클라이언트가 서버에 접속하여 채팅을 주고받는 프로그램</li> <li>메시지는 동기화 됨</li> <li>서버는 실행되어서 클라이언트의 연결을 대기함</li> <li>클라이언트가 연결되면 소켓을 생성한 후, 서버에서 작업할 쓰레드를 생성함</li> <li>서버는 클라이언트의 메시지를 읽고 화면에 뿌려주는 쓰레드로 처리함</li> <li>클라이언트는 쓰레드를 생성하여 서버에서 뿌려주는 메시지를 읽거나 메시지를 보냄</li> </ul>	<a href="http://stellan.tistory.com/entry/Java-Multi-Thread-Net-work-Programming-%EC%97%B0%EC%8A%B5-%EC%B1%84%ED%8C%85-%ED%94%84%EB%A1%9C%EA%B7%B8%EB%9E%A8">http://stellan.tistory.com/entry/Java-Multi-Thread-Net-work-Programming-%EC%97%B0%EC%8A%B5-%EC%B1%84%ED%8C%85-%ED%94%84%EB%A1%9C%EA%B7%B8%EB%9E%A8</a>
웹 서버	<ul style="list-style-type: none"> <li>클라이언트는 URL을 서버에 요청하면 서버는 클라이언트가 요구하는 웹페이지를 제공함</li> <li>서버는 클라이언트의 접속을 대기한 후 연결되면 소켓과 쓰레드를 생성함</li> <li>서버는 URL 메시지를 파싱하여 클라이언트의 요구를 처리함</li> </ul>	<a href="http://theeye.pe.kr/archives/1295">http://theeye.pe.kr/archives/1295</a>
경주마 게임	<ul style="list-style-type: none"> <li>경주마의 수 만큼(6~8) 쓰레드를 생성해서, 정렬 혹은 단순 반복 작업을 수행하게 하여 작업을 끝낸 순서대로 도착한다.</li> </ul>	<a href="http://passionth.tistory.com/273">http://passionth.tistory.com/273</a>
눈치 게임	<ul style="list-style-type: none"> <li>사용자 수 만큼 쓰레드를 생성하여 차례대로 제공되는 범위의 숫자를 요청하여 진행함</li> </ul>	개요 참조 (다음 쪽)

이외에도 멀티쓰레드 예제들이 다수 있음.

## 눈치게임개요

서로의 눈치를 보면서 숫자를 1~ 불러서 마지막에 불리지 못한 사람 혹은 2명 혹은 다수의 사람들이 같은 숫자를 비슷한 시간에 외치는 경우 입니다.

### ○ 눈치게임에서 벌칙을 받게되는 사용자를 골라내는 방법

예) 게임 참가자: A, B, C, D, E (5명), (#) - 벌칙당첨

#### 1. 마지막 번호를 선택하게 되는 경우

- 플레이, 마지막까지 번호가

A: 1!

B: 2!!

C: 3!!!

D: 4!!!!

E: (#)

#### 2. 두 명이 같은 번호를 선택하는 경우

- 같은 번호를 부른 참가자를 포함한 나머지 모든 참가자

- 플레이

A: 1!

B: 2!!

C: 3!!! (#)

D: 3!!! (#)

E: (#)

### ○ 요구 사항

1. 5~10명의 참가자로 구현한다.

2. Java-based Thread Programming

## 눈치게임중 애매한 판정

A~D 중 1,2는 모두 외쳐진 상태에서... A,B 제외

C와 D가 비슷하게 (그러나 확실히 C가 빨리 외쳤습니다.)

그럼 술래는 D가 되는 겁니다. 왜냐하면 마지막 숫자는 먼저 외치는 사람이 승자이기 때문입니다.

이 가정에서 2를 A,B가 외쳤는데 A가 먼저 외쳤다고 하면 술래는 2명이 되는 겁니다.

그 이유는 마지막 숫자가 아니기 때문이죠