

---

## 알고리즘

다이나믹 프로그래밍 : 테이블을 만든다는 뜻, 큰문제를 한번에 해결하기 힘들 때 작은 문제들로 나누어 계산하는 경우가 생긴다 이때 매번 새로 계산하지 않고 값을 저장해 두었다가 재사용하는 기법이 동적프로그래밍 이다.

메모이 제이션도 동적프로그래밍의 하나이다.

그리디 알고리즘 : 각 단계에서 회선의 선택을 하는 기법, 모든경우에 통하는 방법은 아님

Mst : 한버텍스를 기준으로 가능한 한 작은 가중치의 아크들을 사용해서 모든 버텍스를 연결하는 트리를 만드는 것

크루스칼 알고리즘: 선을 기준으로 정렬하는 알고리즘 모든 선을 검색한 뒤 작은선부터 순환이 생기지 않도록 선택

프림 알고리즘: 버텍스를 기준으로 검색하는 알고리즘 한점에서 점점 퍼져나가며 현재의 트리와 연결된 모든 선의 가중치를 이용해 최선의 선택을 하여 트리를 구성하는 방법

다익스트라 알고리즘: 한 버텍스에서 다른 모든 버텍스 까지의 최단경로를 찾는 알고리즘

## Sort 알고리즘

### 그래프 탐색 알고리즘

DFS: 경로를 저장하는데 특화

BFS도 가능하긴 함

BFS: 최단거리를 구해야 하는 문제

네트워크 플로우 알고리즘

최대유량 알고리즘

---

체스게임:

자바 JFrame을 import하여 게임을 만들어 보았다.

Menu와 게임화면을 제작하였으며, 이에 따라 게임의 규칙을 알고리즘을 통해 적용시켜볼 수 있었다.

이벤트 핸들러를 통해 플레이어의 동작을 컴퓨터가 인식하고 이에 따라 말을 움직였다.

말의 생성 및 제거에 class를 활용해 class를 이용한 프로그램 설계에 익숙해졌다.

한계 : 알고리즘이 익숙하지 않을때라 그런지 대각선방향 search 같은경우 매우 더러운 코드였던 걸로 기억한다. 다시하면 정말 빠르게 작성할 자신이 있다.

---

채팅서버 만들기

카카오톡의 메신저 기능을 참고하여 최대한 많은 기능을 지원할 수 있도록 단순히 메신저의 기능을 넘어서 메신저를 통한 개인의 일정관리 시스템을 적용하였다. 자신의 일정을 메신저에 적용해서 사람들과 소통 중 따로 달력을 사용하지 않고도 자신의 일정을 정확히 알고 이후 일정을 정하는것에 도움을 줄 수 있다.

소켓 프로그래밍을 이용하여 작성하였고 서버를 만들기위해 다음과 같은 사항들을 결정하여야 했다.

1. 통신의 패킷을 설정해서 서버와 클라이언트 간 통신메세지의 규칙을 설정해야 한다.
2. 한 서버가 여러 개의 사용자를 수용하기 위해 멀티쓰레드 프로그래밍을 적용하여야 한다.
3. 멀티쓰레드 프로그램이 동작하는 도중 발생할 수 있는 동기화 문제를 해결해야한다.

이러한 것들을 결정하여 프로그램을 제작하였고 한계점으론

1. 통신의 보안문제
2. 쓰레드의 동기화 문제
3. 통신에러의 문제

같은것들이 있었다.

---

블로그 관리:

Html 페이지의 뼈대 그 자체라고 생각한다. 홈페이지의 모든내용을 꾸밈없이 담고있다.

Css 페이지의 뼈대인 Html을 꾸며주기 위하여 작성하였다, 개인적으로 미술적인 감각은 매우 떨어진다고 생각하나 , 다른 코딩블로그의 css 모델을 참고하여 만들었다.

Javascript 위에서 만들어진 잘 꾸며진 글들의 상호작용을 처리하기 위하여 만들었다.

글들에 동작을 넣어 보다 활발한 컨텐츠 제작을 해볼 수 있었다.

Xml http request:

해당 객체는 서버와 상호작용 하기 위하여 사용됨 전체 페이지의 새로고침이 필요없이 URL로부터 데이터를 받아올 수 있습니다.

웹페이지가 사용자의 행위를 방해하지 않으며 페이지의 일부를 업데이트 할 수 있도록 해줌

커뮤니티의 페이지 이동 (1,2,3,4,5 페이지)등이 이런방식으로 작동함.

웹페이지 일부로딩을 통한 네트워크 편의성을 제공해 줄 수 있다.

Ajax programming

페이지의 새로고침 없이 페이지의 내용을 변경해주는 것

웹 브라우저와 웹 서버가 내부적으로 데이터 통신을 하게 된다. 또한 이결과를 웹페이지에 프로그래밍적으로 반영하기 때문에 웹페이지의 로딩 없이 서비스를 사용할 수 있게 됩니다.

Ajax는 Asynchronous javascript and XML의 약자입니다.

자바스크립트를 이용해서 비동기적으로 서버와 브라우저가 데이터를 주고 받는 방식을 의미합니다.

이때 사용하는 API가 XMLHttpRequest입니다.

결과

웹페이지를 html 단위로 구분하며 각 페이지를 anchor를 통해 연결하여 블로그 비슷하게 만들어 보았다.

Css의 박스모델을 이용하여 페이지를 구성하므로써 sidebar,nav, content, header, footer라는 웹페이지의 기본적인 구조에 대하여 생각해 보았다.

Java script 속 XMLHttpRequest를 이용하여 웹 페이지가 직접 서버와 통신하며 비 동기적인 웹페이지 변화를 적용해보았다.

1. 서버에 있는 글 목차를 전부 불러와서 listing 하는 기능 구현
2. List를 정렬하여 버튼을 만든다. 웹 페이지의 페이지징처럼 사용이 가능하도록 코드를 작성하였다.

한계 : 좋은 툴들이 다 구성이 되어있는데 왜 난 자꾸 딴길로 갈까... 방학때 진지하게 툴을 직접 만들거나 아님 툴을 쓰던가 해서 블로그 운영 방식을 바꿔야겠다.

그럼에도 html의 구조에 대해 이해하고 AJAX 통신까지 경험해본 것은 의미있다고 생각한다.

---

## 피자배달 시스템 만들기

비록 학교 과제로 시작했지만 결국 만드는 의미를 찾아버린 경우

전화주문, 방문주문의 이점보다 웹페이지를 통한 피자주문의 이점을 찾는 것이 더 힘들었다.

이 프로젝트에서 나는 review 기능, shop manage 기능을 구현하는 역할을 맡았으며

협업을 하며 든 생각은 역시 대형 프로젝트는 모든 팀원의 생각을 하나로 모으기 위해 큰 시간을 할애하는 것은 절대로 헛된 행동이 아니라는 것이다.

또한 이번 프로젝트에서는 단순히 뭔가를 만드는 것이 아닌 어떤 수요가 있는지에 대한 깊은 고민과 그에 따른 목표 설립, 그리고 sequence와 class diagram을 그리는 것에 대한 체계적인 개발이 인상깊다.

또한 bootstrap과 Django를 동시에 사용하여 여러 간편화된 함수를 실행해보았다.

Class based view 에 대해서 공부하였다.

또한 인터넷 강의의 페이지 구성방식이 내가 직접 만들어보았던 페이지와 많이 다른 것을 확인하였다.

---

## 크롤링

단순히 css를 선택해 가져올 뿐 만 아니라 웹페이지라는 것을 탐색해 볼 수 있을까? 라는 아이디어에서 출발한 기술

어떤 페이지를 크롤링하고 싶을 땐 해당 페이지 메인에 robots.txt를 확인하여야 한다.

예시([naver.com/robots.txt](http://naver.com/robots.txt))

프로토콜?

컴퓨터와 컴퓨터 또는 컴퓨터와 단말기 사이의 통신을 위해 지정하는 규약

포괄적인 의미라고 생각하며 , http ascii jpeg TCP UDP IP IPX Ethernet apple talk 등이 이것에 해당한다.

html

get : 서버에 데이터를 요청하는것입니다.

Post: 서버에 데이터를 생성하는 것을 요청합니다

Put:: 서버에 존재하는 데이터를 수정하거나 존재하지 않으면 생성합니다.

Delete: 서버에 데이터를 제거할것을 요청합니다.

Patch : 요청은 서버에 존재하는 데이터를 일부 수정합니다

.

웹 에러 코드

응답 코드 클래스	의미	설명
1	Informational (정보)	리퀘스트를 받고, 처리 중에 있음.
2	Success (성공)	리퀘스트를 정상적으로 처리함.
3	Redirection (리디렉션)	리퀘스트 완료를 위해 추가 동작이 필요함.
4	Client Error (클라이언트 오류)	클라이언트 요청을 처리할 수 없어 오류 발생함.
5	Server Error (서버 오류)	서버에서 처리를 하지 못하여 오류 발생함.

SQL : 관계형 데이터베이스 관리시스템의 데이터를 관리하기 위해 설계된 특수 목적의 프로그래밍 언어.

관계형 데이터베이스 : key와 value들의 간단한 관계를 테이블화 시킨 간단한 전산정보 데이터베이스이다.

ddl : 데이터 정의어

dml : 데이터 조작어

dcl : 데이터 제어어

다중상속 : 어떤 클래스가 상위 클래스로부터 하나 이상의 특징,행동을 상속받을 수 있는 것을 말한다.

추상클래스 (~이다 의 개념)

다중상속이 불가하다.

인간은 포유동물이다.

인터페이스 (~ 한다 ~ 가능하다 의 개념)

다중상속이 가능하다

~한 동작을 한다는 것을 보장해줄 수 있다.

객체: 어떠한 속성값과 행동을 가지고 있는 데이터

자동차 : 제조사 번호 , 차량 번호 , 차량 크기 ,

앞으로 가기 , 멈추기

객체지향이란? :

의존성 관리를 위한 프로그래밍 기법 , 변경사항을 적용할 때 서로 영향을 최소화하고 독립적인 배포가 가능해집니다. 따라서 독립적인 개발이 가능해지며 고수준 정책과 저수준 구현 세부사항의 분리가 중요합니다.

오버 로딩 : 같은 클래스 내에서 동일한 메소드 이름을 가졌지만 매개변수의 변화로 다르게 동작할 수 있는 것을 의미합니다. 오버라이딩 될 수 있습니다.

오버 라이딩 : 상위 클래스의 함수를 재정의 하는 것을 의미합니다.

String 불변의 변수 문자열을 정의합니다.

String buffer : thread safe 하지 않습니다.

String builder : thread safe 한 함수입니다.

## Deadlock vs Starvation

- 교착상태 : 여러 프로세스가 동일 자원 점유를 요청할 때 발생
- 기아상태 : 여러 프로세스가 부족한 자원을 점유하기 위해 경쟁할 때 발생

## 기아상태 해결 방안

기아상태는 주로 다음과 같은 방법으로 해결한다.

- 프로세스 우선순위 수시 변경을 통해 각 프로세스 높은 우선순위를 가지도록 기회 부여
- 오래 기다린 프로세스의 우선순위 높이기
- 우선순위가 아닌 요청 순서대로 처리하는 요청큐 사용

캐시 란? :

- **캐시 메모리** : 중앙처리장치가 주기억장치에 접근할 때 속도 차이를 줄이기 위해 사용됩니다. 실행 중인 프로그램의 명령어와 데이터를 저장합니다.

기억 용량은 작지만 접근 시간이 주기억장치보다 5 ~ 10 배 정도 빠릅니다.

- **주기억장치** : 중앙처리장치가 직접 데이터를 읽고 쓸 수 있는 장치입니다. 레지스터나 캐시 메모리보다 기억 용량이 큼니다.

자기소개

안녕하십니까 카카오 인턴 서버파트에 지원하게된 지원자 유승훈입니다.

저는 알고리즘을 참 좋아합니다

단순히 코딩테스트를 통과하기 위한 목적이 아닌 문제 해결 능력을 기르기위한 자기개발의 일환으로 보고있으며 이를위해 지난1년간 600문제 이상의 문제를 풀었습니다.

쉽고 간단한 문제를 반복적으로 푸는 것이 아닌 최대한 다양한 알고리즘을 경험하기위해 깊고 넓게 공부하였습니다.

이를통해 문제해결능력이 향상된 것을 느낄 수 있었고, 어떠한 문제를 마주했을 때 할수없다 라는 것 아닌 알고리즘과 데이터구조를 적용해보며 도전해볼 수 있는 지식을 얻었습니다.

또한 2년전 저는 C++을 통해 서버를 제작해본 경험이 있습니다.

소켓프로그래밍, 멀티쓰레드 프로그래밍,그리고 서버에 대하여 공부하였습니다.

아무런 틀없이 시작했기에 이것을 만들었을 때 엄청난 성취감을 느끼며 부족했던 문제해결능력에 대해 아쉬움을 느꼈습니다.

저는 서비스를 만드는것에 많은 흥미를 느끼고있으며 제가 서비스 하고 싶은 아이디어를 직접 제작하기위에 아직 많은 것이 부족하기에 현업에서 많은 것을 배울 기회를 얻고싶었습니다.

따라서 카카오 인턴쉽에 지원하여 현재 많은 분야에 도전중인 카카오에서 다양한 경험을 해보고 싶습니다.



## 이 회사에 지원한 계기

- ➔ 서버를 직접 제작해보며 아무래도 직접 서비스하는 것을 목적으로 만들다 보니 제 프로그램에 부족한부분을 많이 발견하였습니다. 하지만 이러한 부분을 도와줄 수 있는 사람은 주변에 그리 많지 않았습니다. 따라서 직접 서비스를 제작,배포중인 기업에 들어가 현업자들의 지식을 얻을 수 있다면 좋겠다고 생각하였습니다.
- ➔ 또한 카카오는 의견제시가 자유로운 회사라고 생각하였습니다. 저는 어떠한 프로그램을 제작할 때 의견제시를 하는 것을 좋아하는 편입니다. 이를 통해 단순히 코드를 작성하는 경험을 하는 것이 아닌 다양한 시행착오와 문제점을 발견해가며 자기발전할 수 있도록 하고 싶습니다.

## 스트레스 해소법

저는 스트레스를 받으면 주로 단 것을 먹거나 친구에게 전화를 해서 해결합니다. 단 것을 먹으면 기분이 좋아지고 친구에게 전화를 걸어 하소연을 하다보면 그래도 마음이 가벼워지는 것을 느낄 수 있습니다.

## 최근에 관심이 있는 기술

- ➔ 요즘엔 크롤링에 대해 생각하고 있습니다. 이를 통해 어떤 것을 개발할 수 있을까

## 최근에 경험했던 기술적인 결함

- ➔ 어떻게 해결하였는가?

## 커리어 골

- ➔ 10년 , 20년 뒤 목표

## 포트폴리오에서 보고싶은 것?

참고자료, 면접에서 검증(면접의 이야기거리)

## 문제해결능력이 중요함

- 얼마나 고민했는가?
- 어떻게 왜 그방식으로 해결했는가?

개발자 스펙트럼

- 문제해결 능력
- 원만한 협업
- 성장욕구,노력,가능성
- 개발자 문화에 익숙한가?

왜 A를 썼는지 설명해주세요

A에대해 설명해보세요

A와 B의 차이점을 비교해보세요

이 플젝을 하면서 발생한 문제

Git readme를 잘 정리하자

Git 컨벤션을 따랐는가? (디테일)

오픈소스에 기여한 것이 있는가?

코드 커밋 내역?

학점이 낮다면 낮은것에 대한 설명이 필요할 수 있다.

단답이 아닌 면접관에게 자신,자신의 지식에 대한 정보를 전달하려는 노력이 중요하다.

면접관의 질문은 디테일하지 않은경우가 많다, 하지만 나의 대답은 디테일해야한다.