

관계 중심의 사고법

쉽게 배우는 알고리즘

1장. 알고리즘이란

동영상 내용 (<http://vod.kmoocs.kr/vod/2017/08/25/3ab34795-fa79-46b8-9dc9-b8c30bdc5408.mp4>)
알고리즘 수업목표를 이해함

제가 생각하는 알고리즘이 중요한 이유

안녕하세요 실리콘밸리에서 노예 생활을 하고 있는 유저입니다! 제가 생각하는 알고리즘이 (실리콘밸리에 서)중요하다고 생각하는 이유를 2가지 정도 생각나서 적어보겠습니다!

1. 면접 통과!

- 어찌 보면 제일 중요하지 않겠습니까!? 일단 취직을 해야 먹고 살지 않겠습니까!?
- 그럼 왜 면접때 알고리즘 문제를 많이 낼까!? 일반적으로 실리콘밸리에서는 특정 프로젝트에 계약직으로 고용되는 형식이 아니고 풀타임 정규직으로 취직을 하는 형태이면 특정 언어나 기술을 보고 뽑진 않습니다. 그렇기 때문에 모든 사람을 평등(?)하게 평가 할 수 있는 알고리즘을 물어보게 됩니다. 막 어려운 알고리즘을 물어보는게 아니라 소프트웨어 엔지니어라면 알고 있어야 할 기본적인 알고리즘/자료구조를 응용/적절히 사용하는 문제들을 냅니다.

2. 실무에서도 사용한다.

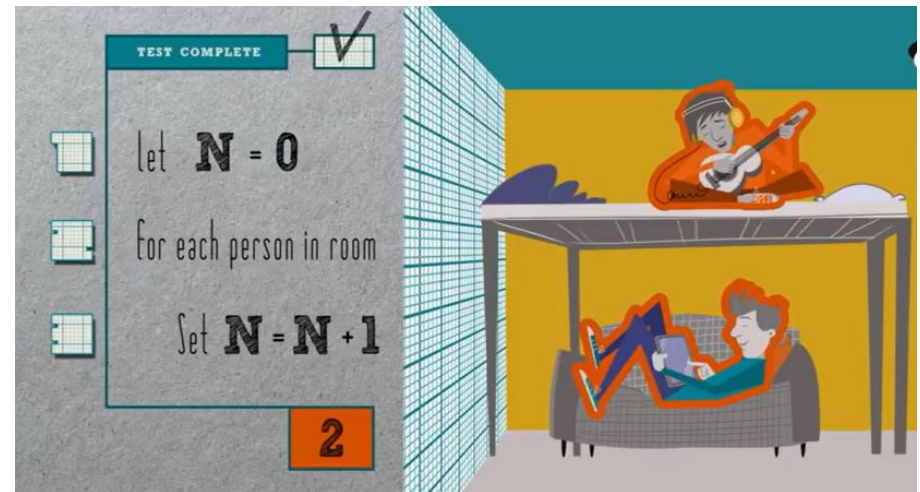
- 실무에서 사용한다고 할때는, 막 링크드리스트/스택/큐 이런걸 일일이 구현한다는 말이 아닙니다. 다만 알고리즘/자료구조/라이브러리등을 이용할때, 최소한 좋은점/나쁜점/런타임BigO/메모리BigO 를 이해하면서 사용해야 한다는 말입니다.

저는 개인적으로 알고리즘/자료구조는 소프트웨어 엔지니어라면 기본 소양이라고 생각해서(그리고 언젠가의 이직을 위해서 ㅋㅋ) 보통 1주일에 새롭게 등장하는 문제들 1~3개정도씩은 꾸준히 풀고 있습니다.

알고리즘이란

- 정의
 - 주어진 문제를 해결하기 위한 단계 혹은 절차
 - 유한한 단계를 거쳐서 반드시 종료
 - 문제 해결을 위한 모호하지 않은 명령어의 순서 (sequence of unambiguous instructions)
- “Algorithms” + Data Structures = Programs
- 아라비아의 천문학자, 지리학자, 수학자인 알콰리즈미 (al-Khwarizmi)의 이름에서 유래하고 있음

알고리즘이란



https://www.tutorialspoint.com/online_c_compiler.php

알고리즘은 문제 해결 과정을 묘사하는 것

- 문제 해결 절차를 체계적으로 기술한 것
- 문제의 요구 조건
 - 입력과 출력으로 명시할 수 있다
 - 알고리즘은 입력으로부터 출력을 만드는 과정을 기술



입출력의 예

- 문제
 - 100명의 학생의 시험점수의 최대값을 찾으라
- 입력
 - 100명의 학생들의 시험점수
- 출력
 - 위 100개의 시험점수들 중 최대값

알고리즘 공부의 목적

- 특정한 문제를 위한 알고리즘의 습득
- 체계적으로 생각하는 훈련
- 지적 추상화의 레벨 상승

추상화 (컴퓨터 과학)

위키백과, 우리 모두의 백과사전.

컴퓨터 과학에서 추상화(abstraction)는 복잡한 자료, 모듈, 시스템 등으로부터 핵심적인 개념 또는 기능을 간추려 내는 것을 말한다.

알고리즘은 생각하는 방법을 훈련하는 것

- 문제 자체를 해결하는 알고리즘을 배운다
- 그 과정에 깃든 ‘생각하는 방법’을 배우는 것이 더 중요하다
- 미래에 다른 문제를 해결하는 생각의 빌딩블록을 제공한다



알고리즘은 자료구조의 확장



자료구조



빠른 경로 찾기를 이용할 때는 그래프가 활용될 수도 있습니다



따라 달라지는 것이라고



알고리즘은 집을 짓는 과정들이 알고리즘이라고 이야기 할 수 있겠습니다



자료구조를 잘 파악하고 적합한 알고리즘을 찾아내는 것

알고리즘은 자료구조의 확장

- 선행 과목



알고리즘 공부

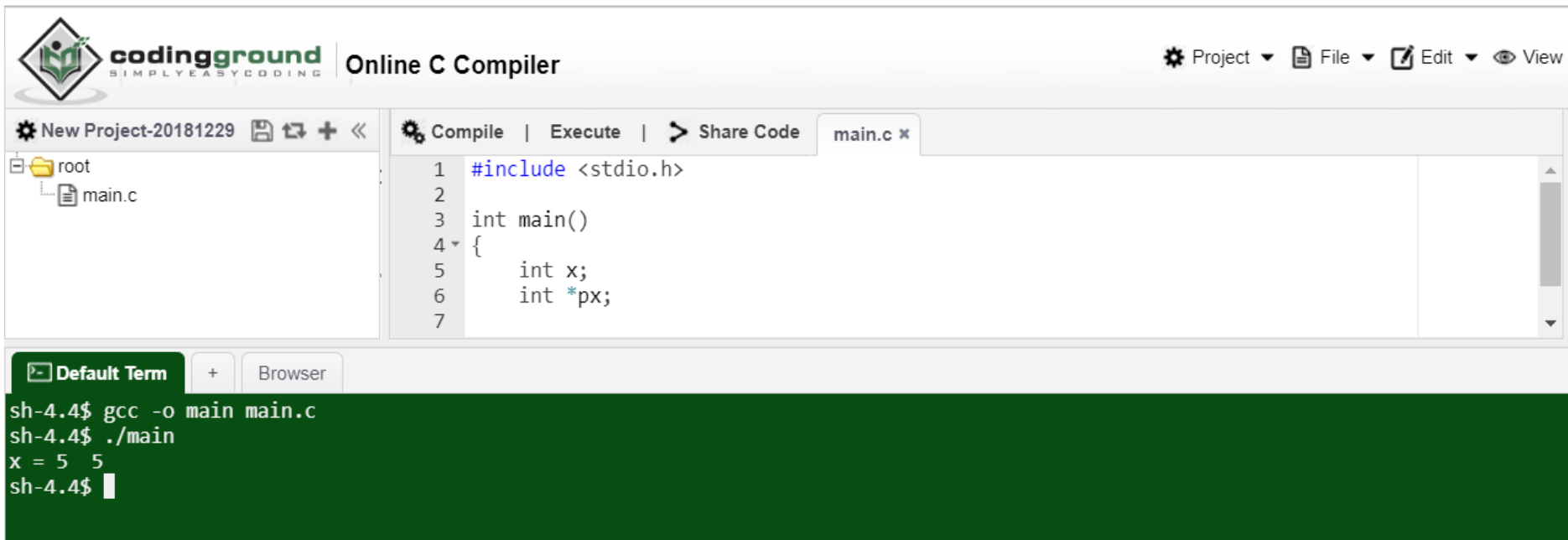
- 많은 사람들이 공통적으로 얘기하는 사실
 - 처음부터 어려운 걸 하려고 하지 말고, 간단한 것부터 시작
 - 선택한 언어의 문법과 클래스를 잘 파악
 - 풀고 난 후 다른 사람의 풀이 참고
 - 경험이 쌓이면 익숙해 짐, 조금 해하지 말자



Thank you

온라인 프로그램 작성

https://www.tutorialspoint.com/online_c_compiler.php



The screenshot displays the CodingGround Online C Compiler interface. At the top, the logo for 'codingground' is visible, along with the text 'SIMPLY EASY CODING' and 'Online C Compiler'. The interface includes a menu bar with 'Project', 'File', 'Edit', and 'View' options. Below the menu bar, there are tabs for 'New Project-20181229', 'Compile', 'Execute', and 'Share Code'. The 'main.c' file is open in the editor. The code in the editor is as follows:

```
1 #include <stdio.h>
2
3 int main()
4 {
5     int x;
6     int *px;
7 }
```

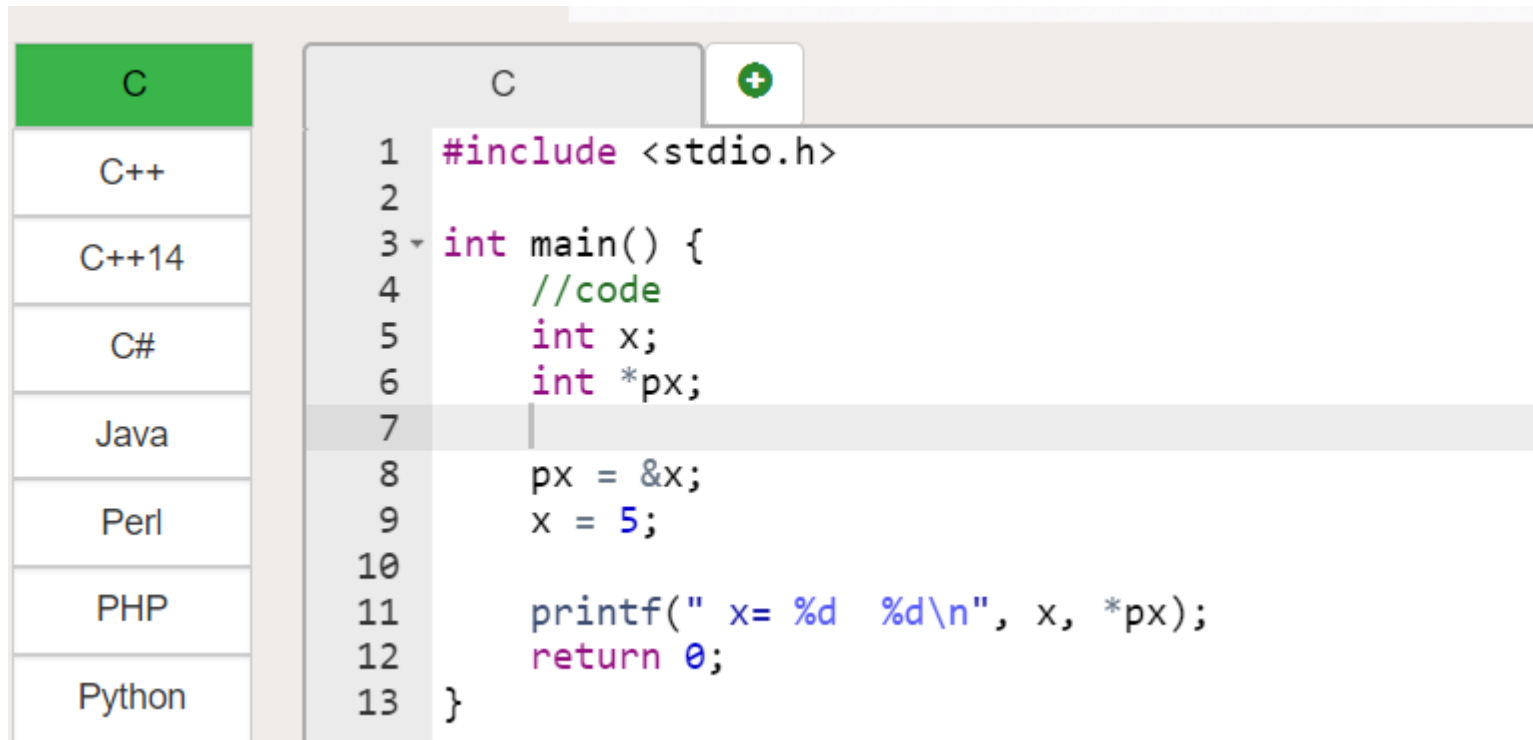
Below the editor, there is a terminal window showing the execution of the program. The terminal output is as follows:

```
sh-4.4$ gcc -o main main.c
sh-4.4$ ./main
x = 5 5
sh-4.4$
```

gcc -o main main.c

온라인 프로그램 작성

<https://ide.geeksforgeeks.org>



The screenshot shows the GeeksforGeeks online IDE interface. On the left, there is a sidebar with a list of programming languages: C (highlighted in green), C++, C++14, C#, Java, Perl, PHP, and Python. The main editor area displays a C program with the following code:

```
1  #include <stdio.h>
2
3  int main() {
4      //code
5      int x;
6      int *px;
7
8      px = &x;
9      x = 5;
10
11     printf(" x= %d  %d\n", x, *px);
12     return 0;
13 }
```

알고리즘 평가

- 어떠한 알고리즘이 좋은가 ?
- 간단한 방법: 알고리즘 구현, 수행, 시간 측정하고 측정 값을 평균함
 - 장점
 - 시스템의 성능에 미치는 영향 정도를 발견함
 - 수학적 계산이 필요 없음
 - 단점
 - 프로그램을 해야 함
 - 성능 측정이 어려운 경우 알고리즘 평가가 쉽지 않음
 - 두 개 알고리즘 비교시 공정한 평가를 위해 동등한 조건을 만들어야 함(동일 컴퓨터, 동일 OS, 동일한 시스템 상황)

알고리즘 평가

- 작은 입력은 알고리즘의 효율과 상관없이 빨리 끝남
- 입력의 크기가 매우 큰 경우(예, 빅데이터)
- 입력의 크기에 따라 함수가 증가하는 비율을 점근적 증가율이라고 하며, 점근적 표기방법으로 복잡도를 계산함