

일반화 되었다 -> input이 달라져도 알고리즘은 그대로 있다.

십진수 d를 이진수 b로 바꾸는 알고리즘 d2b?

알고리즘에서 좀더 명확한 방법을 쓰자 : "변수"를 쓰자는 것이다.

변수에 수를 계속 집어 넣는 것이다.

1 d2b(d) : input이 d

2 while(d != 0){

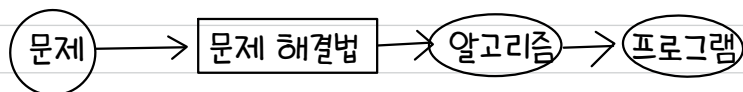
3 r = d/2의 나머지

...

7 return b : output이 b

아무것도 모르는 사람이 이 알고리즘만 따라가면 십진수를 이진수로 바꾸는 것이 가능!

알고리즘을 만든 뒤 프로그래밍을 한다.



프로그램에는 컴퓨터가 이해가능한 명령어가 이미 정의되어 있다.

정해진 명령어들만 사용해서 명확하게 표현.

컴퓨터 프로그래밍 언어 : 컴퓨터가 이해가능한 언어

프로그래밍 언어 -> 기계어 -> 하드웨어 통제

사실 우리가 기계어를 알 필요는 없다.

컴퓨터는 미리 정해진 명령어들만 수행이 가능하다.

프로그래밍 언어가 human-friendly인지, computer-friendly인지가 다르다.

알고리즘을 만드는 과정

1. 예전에 문제를 해결한 적이 있다면

-> 그것을 정리하면 된다.

가장 빠른 계산대를 가기 위한 알고리즘?

-> 줄길이, 물건의 양, 나와의 거리, 출구와의 거리, 계산원의 속도, 줄서있는 사람의 성향

사람의 얼굴을 인식할 수 있는 문제 (자기가 정확히는 모름)

내 머릿속에는 그것을 판단하는 알고리즘이 있다.

당구공의 무게를 같게하기

traveling sales man

하나를 검사하는데 1초가 걸리고 $n = 100$ 이면 $2.9 * 10^{148}$ 년이 걸린다.

이것이 알고리즘의 복잡도 문제, 알고리즘은 확장성 문제이다.

알고리즘은 정확하지만, 쓸 수 없는 알고리즘이다.

-> possible but practically impossible

스크래치 : MIT의 교육용 프로그램

Lego와 비슷하게 하였다.

프로세싱 : 미디어 연구소에서 만들었음

파이썬 : 이 정도는 알아 놓자

문장을 reverse시키는 문제

배열이라는 구조를 이용하면 그 문제를 쉽게 해결가능하다.

A[1] -> B[5]