

Discrete Mathematics HW 4: Boolean Function Solver (~12/10 24:00)

목표

본 과제에서는 Boolean algebra $\langle B, *, +, ', 0, 1 \rangle$ 와 $f: B^2 \rightarrow B$ 가 주어졌을 때 f 가 boolean function인지 판별하고 만약 그렇다면 해당하는 boolean expression이 무엇인지 알아내는 프로그램을 작성하는 걸 목표로 합니다. 아래에서 요구하는 함수들을 포함하는 한 개의 .py 파일을 제출해야 합니다.

배경지식

Stone's representation theorem에 따라 임의의 boolean algebra는 어떤 atom들의 power set algebra로 표현할 수 있습니다. 이 프로그램에서는 atom을 서로 다른 string으로 정의하고, 모든 atom들을 list에 담은 방식으로 boolean algebra를 표현합니다. 엄밀하게는 다음과 같습니다.

- Atom: string (예. 'a')
- Boolean algebra: list of Atoms (예. ['a', 'b', 'c'])
- Element (of B): set of Atoms (예. {'a', 'c'})

한 편, 어떤 $f: B^2 \rightarrow B$ 이 boolean function이라면 두 개의 boolean variable(이 과제에서는 x, y 로 칭합니다)로 이루어진 어떤 boolean expression g 가 존재하여 모든 $(x, y) \in B^2$ 에 대해 $f(x, y) = g(x, y)$ 가 성립해야 합니다. 따라서 주어진 f 가 boolean function인지를 판별하기 위해서는 가능한 모든 boolean expression g 에 대해 f 와 g 를 비교해야 합니다.

이 때, 임의의 boolean expression은 minterm들의 합의 꼴로 환원 가능함이 알려져 있습니다. 여기서 minterm이란 boolean variable x_1, x_2, \dots, x_n 이 있을 때 $(x_1$ 과 $x_1')$, $(x_2$ 와 $x_2')$, ..., $(x_n$ 과 $x_n')$ 중 각각 하나씩 골라 * 연산자로 이은 것을 가리킵니다. 두 개의 boolean variable x, y 에 대해서는 다음과 같이 총 4종류의 minterm이 있습니다. (편의상 * 연산자는 생략하여 표시하였습니다.)

$$xy \quad x'y \quad xy' \quad x'y'$$

예를 들어, $g(x, y) = x + (x + y)x'y$ 라고 하면 $g(x, y) = x + xx'y + yx'y = x + 0y + x'y = x + x'y = x(y + y') + x'y = xy + xy' + x'y$ 와 같이 minterm들의 합의 꼴로 환원시킬 수 있습니다.

과제 목표

본 과제에서는 다음과 같은 함수를 구현하는 것이 목표입니다.

```
find_boolean_function(atoms, func)
```

- 입력

- atoms: 주어진 boolean algebra의 모든 Atom들 (list of string)
 - 예) ['a', 'b', 'c']
- func: Element 두 개를 입력으로 받아 Element를 반환하는 함수
 - 예) func({'a', 'b'}, {'c'})는 {'a', 'c'}를 반환함
- 반환
 - 만약 func가 boolean function이라면
 - 상응하는 boolean expression을 minterm들의 합의 꼴로 환원시켰을 때의 minterm들을 list에 담아 반환합니다.
 - List 내에서 minterm의 순서는 상관 없지만, 중복되는 minterm이 있으면 안됩니다.
 - Minterm은 string으로 다음의 4가지 중 하나여야 합니다: "xy", "x'y", "xy'", "x'y'"
 - 예) ["xy", "xy'"]
 - $f(x, y) = 0$ 인 경우 빈 리스트를 반환하면 됩니다.
 - 만약 func가 boolean function이 아니라면 None을 반환합니다.

채점

총 100개의 테스트 케이스에 대해 채점을 진행합니다. 각 테스트케이스의 배점은 1점이며, 당신의 find_boolean_function 함수가 test input에 대해 올바른 output을 반환할 경우 점수가 부여됩니다.

그러나 find_boolean_function 함수의 구현이 Python 프로그래밍을 처음 접한 경우에는 다소 복잡할 수 있어 단계별 구현 및 부분점수를 적용할 수 있도록 [뼈대코드](#)를 준비했습니다. 뼈대코드에서 find_boolean_function은 이미 구현이 완료되어 있으며, 아래의 4가지 함수를 올바르게 구현하시면 동작합니다.

각 테스트케이스에 대해 만약 find_boolean_function 함수가 올바른 결과를 반환하지 않았다면 아래의 4개의 함수 중 몇 개가 올바른 결과를 반환하는지를 바탕으로 0.25점, 0.5점, 혹은 0.75점의 부분점수를 부여합니다. (뼈대코드를 사용하여 구현하지 않으셨더라도 아래의 함수들을 스펙에 맞게 추가로 구현하시면 채점시 마찬가지로 부분점수가 부여되나, 뼈대코드 사용을 권장드립니다.)

zero_and_one(atoms)

- 입력
 - atoms: 주어진 boolean algebra의 모든 Atom들 (list of string; nonempty)
- 반환
 - 주어진 boolean algebra에서 0과 1에 해당하는 element를 list의 형태로 반환
 - 즉, [zero, one]의 형태

`list_all_elements(atoms)`

- 입력
 - atoms: 주어진 boolean algebra의 모든 Atom들 (list of string; nonempty)
- 반환
 - 주어진 boolean algebra의 모든 Element들을 담고 있는 list (0과 1도 포함)
 - List 내에서 Element의 순서는 중요하지 않음
 - 그러나 중복된 Element가 있으면 안됨

`evaluate_minterm(x, y, one, minterm)`

- 입력
 - x: Element (set type)
 - y: Element (set type)
 - one: 주어진 boolean algebra에서 1에 해당하는 Element (set type)
 - minterm: "xy", "x'y", "xy'", "x'y'" 중 하나 (string type)
- 반환
 - 주어진 x와 y를 이용해서 주어진 minterm의 값을 계산한 결과 Element

`add_elements(x, y, zero, elements)`

- 입력
 - x: Element (set type)
 - y: Element (set type)
 - zero: 주어진 boolean algebra에서 0에 해당하는 Element (set type)
 - elements: 더해야 할 Element들의 리스트 (list of set)
- 반환
 - elements 안의 모든 Element를 합(+)한 결과 Element를 반환
 - elements가 빈 리스트인 경우 0에 해당하는 Element를 반환