

## Algorithm homework 2

공과대학 산업공학과

2018-10786 서규혁

### 1. Explain how your checker program works in your report.

checker program은 input과 output txt 파일을 담고있는 위치를 입력받아 작동된다. 이는 checker라는 함수로 checker함수를 부르면, input\* 이름을 가지는 directory안에 input.txt 와 output.txt를 readlines를 통해 읽어온다. output.txt에는 input.txt의 내용과 각 instruction별 결과가 뒤이어 나오므로 그 길이가 input.txt를 readlines 해온 길이의 2배여야 한다. 이를 체크하고 2배가 아니면 바로 result에 False를 입력한다. 만약 2배인 경우, input.txt의 내용과 output.txt에서 앞의 input.txt와 겹치는 부분을 제외한 내용을 compare함수의 입력값으로 넣어준다.

compare함수를 보자. compare함수는 우선 input의 길이, 즉 instruction의 갯수를 l(영어 L)로 지정하고 길이가 9999이고 각 element가 0인 리스트 A를 만든다. 그리고 문자열 result의 초기 값을 True로 설정한다. 그리고 각 instruction을 했을 때 그 결과가 제대로 나타났는지 살펴볼 것이며 그렇지 않은 경우를 발견하면 그 즉시 False라는 문자열을 return 할 것이다.

instruction이 I로 시작하는 경우를 살펴보자. 만약 'I n' 인 경우, 즉 n을 insert하는 경우에  $A[n-1]==0$  이면 n은 현재 Tree에 없으므로  $A[n-1]$ 에 1을 입력한다. 그리고 해당 instruction의 결과값이 n가 아닌 경우에는 그 즉시 False문자열을 리턴하게 된다. 만약  $A[n-1]==1$ 이면 n은 현재 Tree에 있다는 뜻이므로  $A[n-1]$ 에 0을 입력하며 해당 instruction의 결과값이 0이 아니라면 그 즉시 False 문자열을 리턴한다. 그 외는 정상인 경우이므로 다음 instruction을 살핀다.

Instruction이 D로 시작하는 경우를 보자. 만약 'D n' 인 경우, 즉 n을 delete하는 경우에  $A[n-1]==1$  이면 n은 현재 Tree에 있으므로  $A[n-1]$ 에 0을 입력한다. 그리고 해당 instruction의 결과값이 n가 아닌 경우에는 그 즉시 False문자열을 리턴하게 된다. 만약  $A[n-1]==0$ 이면 n은 현재 Tree에 없다는 뜻이고 지울 대상이 없으므로 별다른 작업없이, 해당 instruction의 결과값이 0이 아니라면 그 즉시 False 문자열을 리턴한다. 그 외는 정상인 경우이므로 다음 instruction을 살핀다.

Instruction이 S로 시작하는 경우를 살펴보자. 만약 'S n' 인 경우, 즉 n번째 element를 찾아달라고 하는 경우인 것이다. 반복문을 돌게 될 j를 처음에 1로 설정하고,  $A[0:j]$  즉 1부터 j-1까지의 숫자가 해당 Tree에 몇 개 있는지 살펴본다. 만약 n-th element를 찾을 수 없다면, 즉 Tree에서 관리하는 수의 갯수가 찾고자 하는 n-th의 n보다 작은 경우, instruction의 결과값이 0이 아니면 문자열 False를 리턴한다. 만약 n-th element를 찾을 수 있다면, 즉  $\text{sum}(A[0:j])==n$ 이 되는 최초의 순간은 당연히  $A[j-1]$ 이 0이 아닌 1이기 때문이고 이 뜻은 j이하의 수가 Tree에 n개 있다는 말, 즉 j가 n-th element라는 것이다. 그래서 instruction의 결과가 j가 아니게 되면 그 즉시 문자열 False를 리턴한다. 그 외는 정상인 경우이므로 다음 instruction을 살핀다.

Instruction이 R로 시작하는 경우를 살펴보자. 만약 'R n' 인 경우,  $A[n-1]$ 을 방문한다. 만약 이 값이 0이라면 n은 Tree에 없는 값이므로 이 instruction의 결과가 0이 아니면 그 즉시 False 문자열을 리턴한다. 만약 이 값이 1이라면, n이 몇번째 element인지 파악해야하므로  $\text{sum}(A[:n])$

을 하게 되고 이 값이 instruction의 결과값과 다르면 그 즉시 False를 리턴한다. 그 외는 정상  
경우이므로 다음 instruction을 살핀다.

이 외의 문자로 시작하는 Instruction의 경우 잘못 되었으므로 False가 된다.

이렇게 모든 instruction에 대해 True False를 비교한 뒤 모든 케이스를 False 없이 마치게 되면  
compare의 리턴값은 True가 되며 이를 저장하여 checker.txt 에 write하는 것이다.

## 2. Write down the environment you run your program and how to run your program.

작업환경은 python 3.8.9이며 glob을 import했다.

해당 프로그램을 run하는 방법은, terminal을 켜고 main.py가 있는 경로에 접속한다. 해당 경로  
에서 \$ python main.py 를 하면 main.py가 실행된다.

## 3. Example Running

아래 표는 source code 인 main.py 와 같은 경로에 있는 directory input1 안에 저장된 3 개의 txt  
파일로 Example running 이다.

input.txt	output.txt	checker.txt
<div>1 I 17</div> <div>2 I 3</div> <div>3 I 22</div> <div>4 I 44</div> <div>5 I 19</div> <div>6 I 21</div> <div>7 I 6</div> <div>8 I 10</div> <div>9 D 4</div> <div>10 D 22</div> <div>11 I 22</div> <div>12 I 3</div> <div>13 S 2</div> <div>14 S 10</div> <div>15 R 5</div> <div>16 R 3</div> <div>17</div>	<div>1 I 17 17 17</div> <div>2 I 3 18 3</div> <div>3 I 22 19 22</div> <div>4 I 44 20 44</div> <div>5 I 19 21 19</div> <div>6 I 21 22 21</div> <div>7 I 6 23 6</div> <div>8 I 10 24 10</div> <div>9 D 4 25 0</div> <div>10 D 22 26 22</div> <div>11 I 22 27 22</div> <div>12 I 3 28 0</div> <div>13 S 2 29 6</div> <div>14 S 10 30 0</div> <div>15 R 5 31 0</div> <div>16 R 3 32 1</div>	<div>1 True</div>

아래 표는 source code 인 main.py 와 같은 경로에 있는 directory input2 안에 저장된 3 개의 txt 파일로 Example running 이다.

input.txt		output.txt				checker.txt
1	I 1	1	I 1	19	1	1   True
2	I 2	2	I 2	20	2	
3	I 3	3	I 3	21	3	
4	I 4	4	I 4	22	4	
5	I 5	5	I 5	23	5	
6	I 100	6	I 100	24	100	
7	I 100	7	I 100	25	0	
8	D 100	8	D 100	26	100	
9	D 100	9	D 100	27	0	
10	D 100	10	D 100	28	0	
11	R 100	11	R 100	29	0	
12	S 10	12	S 10	30	0	
13	S 1	13	S 1	31	1	
14	S 5	14	S 5	32	5	
15	R 1	15	R 1	33	1	
16	R 3	16	R 3	34	3	
17	R 5	17	R 5	35	5	
18	R 0	18	R 0	36	0	
19						