

6. [2 marks] Your task is to write a Python program that can take a natural deduction proof (as a text file), and output *correct* or *incorrect* depending on whether the proof is correct or not.

The first line of the input proof file will always be the sequent whose validity is being proved. The second line will be an empty line, and the proof will begin from the third line. Each line of the proof will begin with an explanation preceeding the proof statement (this is unlike the proofs we did in the class, where we wrote the explanations *after* the statements). The explanations can only be of the following forms (where i, j, k, l, m, n denote line numbers in the proof file, as expected): [premise], [assumption], [copy i], [mp i, j], [mt i, j], [and-in i, j], [and-e1 i], [and-e2 i], [or-in1 i], [or-in2 i], [or-e1 $i, j-k, l-m$], [impl-in $i-j$], [neg-in $i-j$], [neg-e1 i, j], [bot-e1 i], [dneg-in i], [dneg-e1 i], [pbc $i-j$], [lem].

The keywords are self-explanatory, and the precise interpretation of these rules can be found in Chapter 1 of the book by Huth and Ryan. Please also note that we will denote the logical symbols \wedge and \vee using backward and forward slashes, \neg using the ! symbol, \rightarrow using \rightarrow , and \vdash using |- in the text file.

Here are two sample input files and their corresponding output:

$q \rightarrow r \vdash p \vee q \rightarrow p \vee r$

[premise]	$q \rightarrow r$
[assumption]	$p \vee q$
[assumption]	p
[or-in1 5]	$p \vee r$
[assumption]	q
[mp 7, 3]	r
[or-in2 8]	$p \vee r$
[or-e1 4, 5-6, 7-9]	$p \vee r$
[impl-in 4-10]	$p \vee q \rightarrow p \vee r$

Output: correct

$p \rightarrow q \vdash !p \vee q$

[premise]	$p \rightarrow q$
[lem]	$q \vee !q$
[assumption]	q
[or-in2 5]	$!p \vee q$
[assumption]	$!q$
[mp 7, 3]	$!p$
[or-in1 8]	$!p \vee q$
[or-e1 4, 5-6, 7-9]	$!p \vee q$

Output: incorrect