Software Verification 2021/22

Proof Strategies for Backward Reasoning

- 1. If your goal is $\neg A$ then
 - (a) Do $\lceil \neg I \rceil$; now it is enough to prove \bot using A as an assumption. This looks like : $\frac{\perp}{\neg A} [\neg I]$
- 2. If your goal is $A \to B$ then
 - (a) Do $[\rightarrow I]$; now it is enough to prove B using A as an assumption. This looks like : $\frac{B}{A \to B} [\to I]$
- 3. If your goal is $A \wedge B$ then
 - (a) Do $[\land I]$; now it is enough to prove both A and B separately. This looks like : $\frac{A}{A \wedge B} [\land I]$
- 4. If your goal is $A \vee B$ then
 - (a) Do $[\vee I]$ choosing one of A or B to prove which one to choose is not always clear! Looking at the given hypotheses or the current context might give a clue as to which one to prove. **Warning**: Choosing the wrong option to prove can lead to a dead-end! This looks like : $\frac{A}{A \vee B} [\vee I]$ or $\frac{B}{A \vee B} [\vee I]$

- 5. If your goal is A then
 - (a) Try $[\wedge E]$ if there is a hypothesis/context of the form $A \wedge B$ This looks like : $\frac{A \wedge B}{A} [\wedge E]$

 - (c) Try $[\lor E]$ if there are hypotheses/contexts of the form $X \lor Y, \ X \to A$ and $Y \to A$ This looks like : $X \lor Y \longrightarrow X \longrightarrow A \longrightarrow A \longrightarrow A \longrightarrow A$
- 6. If your goal is \perp then
 - (a) Try $\neg E$ if there are hypotheses/contexts of the form Z and $\neg Z$ This looks like: $Z = \overline{Z} = \overline{Z}$ [$\neg E$] Note that this is a special case of 5b above.

Software Verification 2021/22

In addition to the aforementioned strategies, for predicate logic we have the following additional strategies:

- 7. If your goal is $\neg \forall x, P(x)$ then
 - (a) Do $[\neg I]$; now it is enough to prove \bot using $\forall x, P(x)$ as an assumption. This looks like : $\frac{\bot}{\neg \forall x, P(x)} [\neg I]$
- 8. If your goal is $\neg \exists x, P(x)$ then
 - (a) Do $[\neg I]$; now it is enough to prove \bot using $\exists x, P(x)$ as an assumption. This looks like : $\frac{\bot}{\neg \exists x, P(x)} [\neg I]$
- 9. If your goal is $\forall x, P(x)$ then
 - (a) Try $[\forall I]$; now it is enough to prove P(t) for a **general** variable t in the domain This looks like : $\frac{P(t)}{\forall x, P(x)} [\forall I]$
- 10. If your goal is $\exists x, P(x)$ then you need to supply an element t: U of the domain and a proof of P(t), i.e., show that t satisfies P.
 - (a) The t:U to be supplied could be a constant (e.g., $2:\mathbb{N}$) or a variable already in the context (e.g., some $n:\mathbb{N}$). Use $[\exists I]$ supplying a suitable element a:U. This looks like: $\frac{P(a)}{\exists x, P(x)} [\exists I]$
- 11. If your goal is P(t) for a **general** variable t in the domain then check your hypotheses to see what might allow to prove P(t). Eliminate that hypothesis/these hypotheses, using the corresponding elimination rule.

The two hardest (most non-intuitive) rules to apply are $[\vee E]$ and $[\exists E]$. In general, if you are stuck and don't see any way to proceed with your proof, then maybe consider the following three things:

- 1. Try $[\vee E]$
- 2. Try $[\exists E]$
- 3. Try either DNE or EM from classical logic. **Warning**: Do not use rules from classical logic if the question explicitly says to only use rules from intuitionistic logic only.