

# Project 2: The **Static Semantics** of SPL

## Part 2a: **Scoping-Analysis**

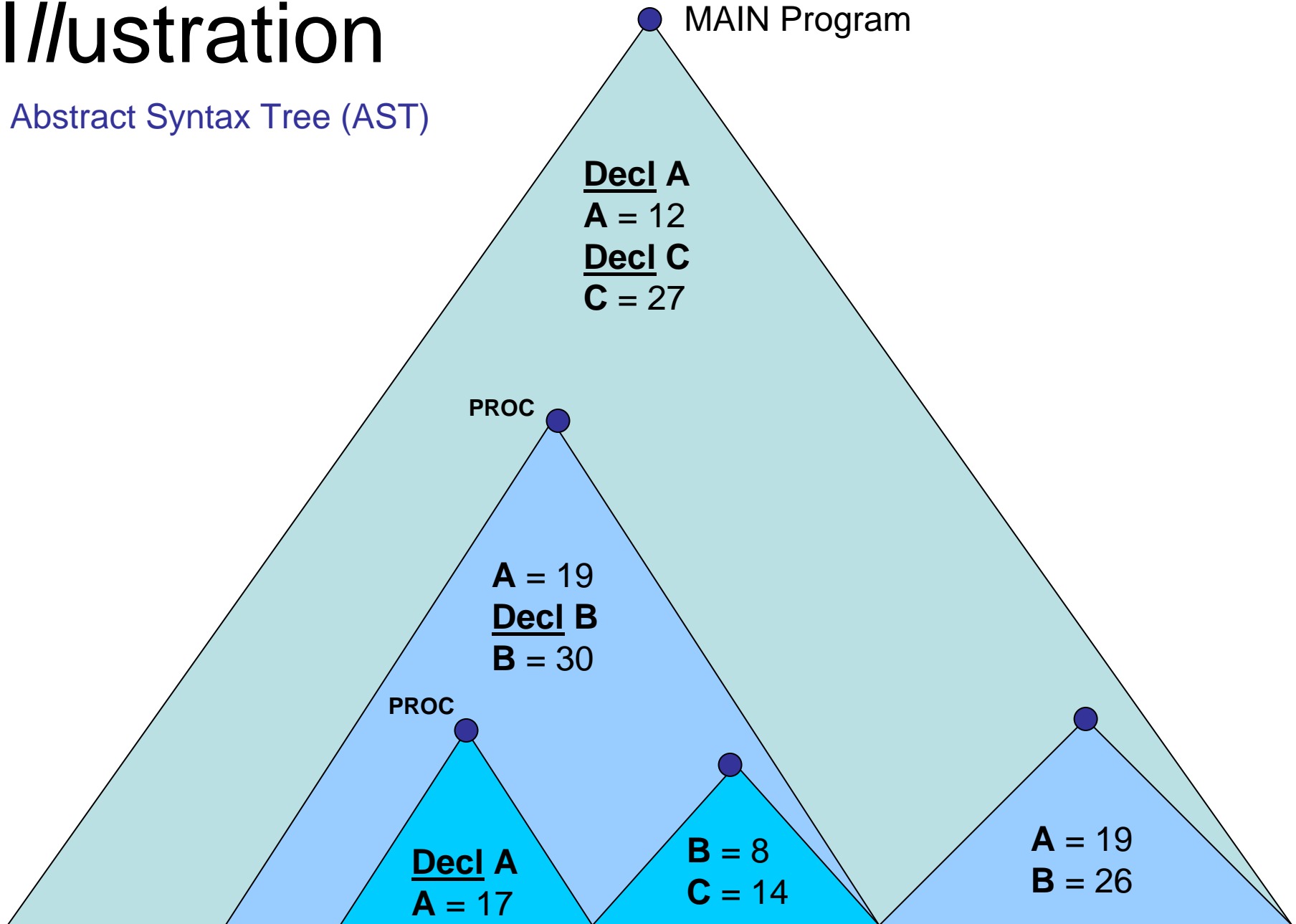
*with New Project Partner* (😊😊)

# Scoping Rules for SPL

- are **defined** as follows:
  - The “Main Program” opens the outermost Name-Scope (**Level 0**),
  - Every Procedure Declaration opens an inner Name-Scope (**Level  $n+1$** ),
  - Visibility of Declarations is “inside-out” or “upwards” (from Level  **$n$**  to Level  **$n-1$** ).
    - This is for **both**
      - User-defined **Variable** Names,
      - User-defined **Procedure** Names

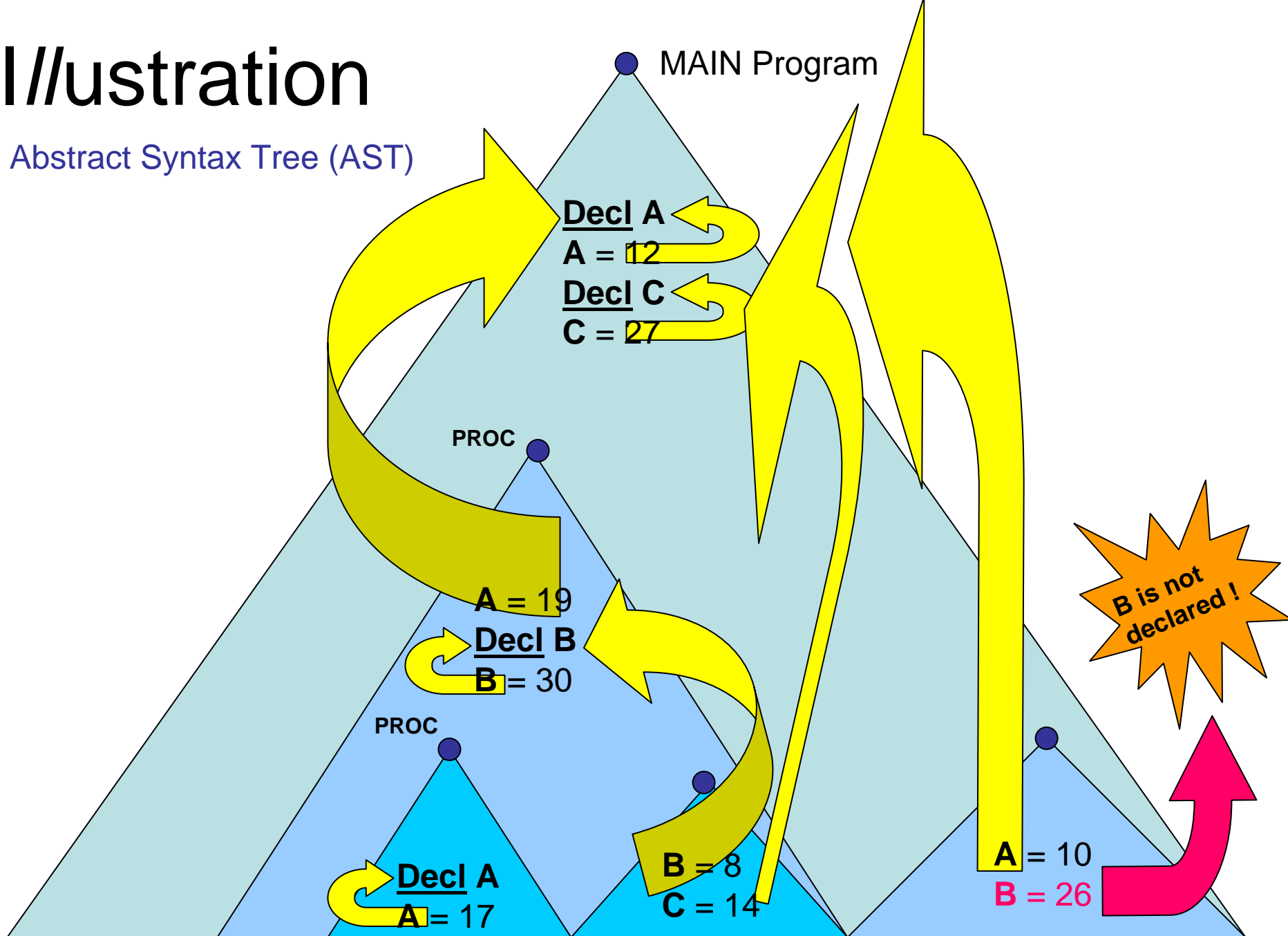
# Illustration

Abstract Syntax Tree (AST)



# Illustration

## Abstract Syntax Tree (AST)



# Your **Tasks** for Sub-Project 2a:

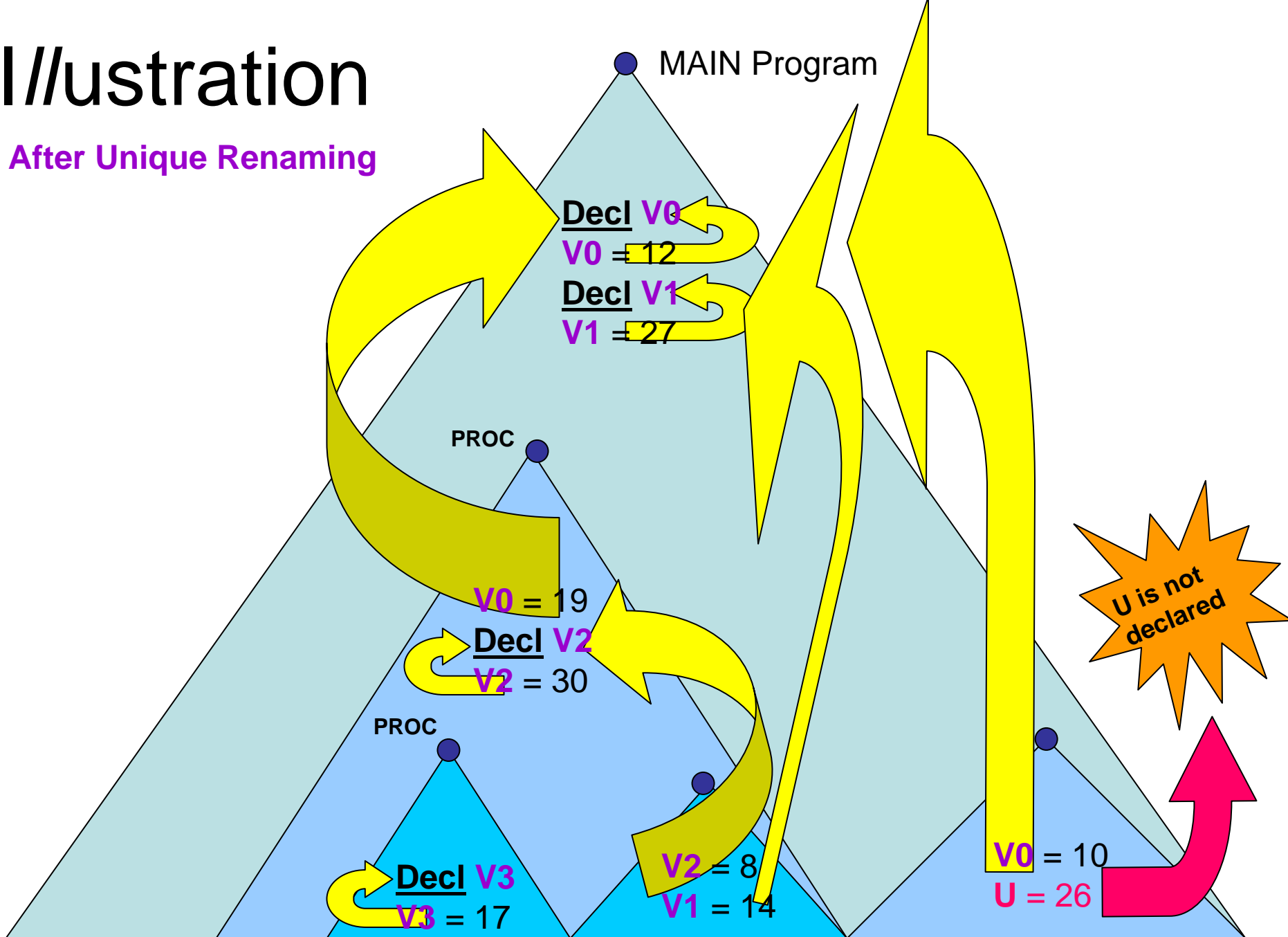
- **Analyse the AST data structure** which your Parser had generated in Project 1b.
- Write an **algorithm that “crawls” up and down the AST**:
  - To each Node in the AST, attach a **Scope-ID** in the **Semantic Information Table**.
  - To each Node which represents the **usage** of a name (**var-name** or **proc-call**), determine its corresponding **declaration** Node.
  - “Populate” the Semantic Table accordingly →

# Your **Tasks** (continued):

- Note: Every AST Node has a “pointer” into the Semantic Table!
- After you have “populated” the Semantic Table with Scope\_IDs for each Node, as well as the information about the location of a Declaration for each “used” Name:
  - Apply a consistent re-naming of all occurring names with new “fresh” and unique names:
    - For variable names: **V0**, **V1**, **V2**, **V3**, ...
    - For procedure names: **P0**, **P1**, **P2**, **P3**, ...
    - For any un-declared name: **U**

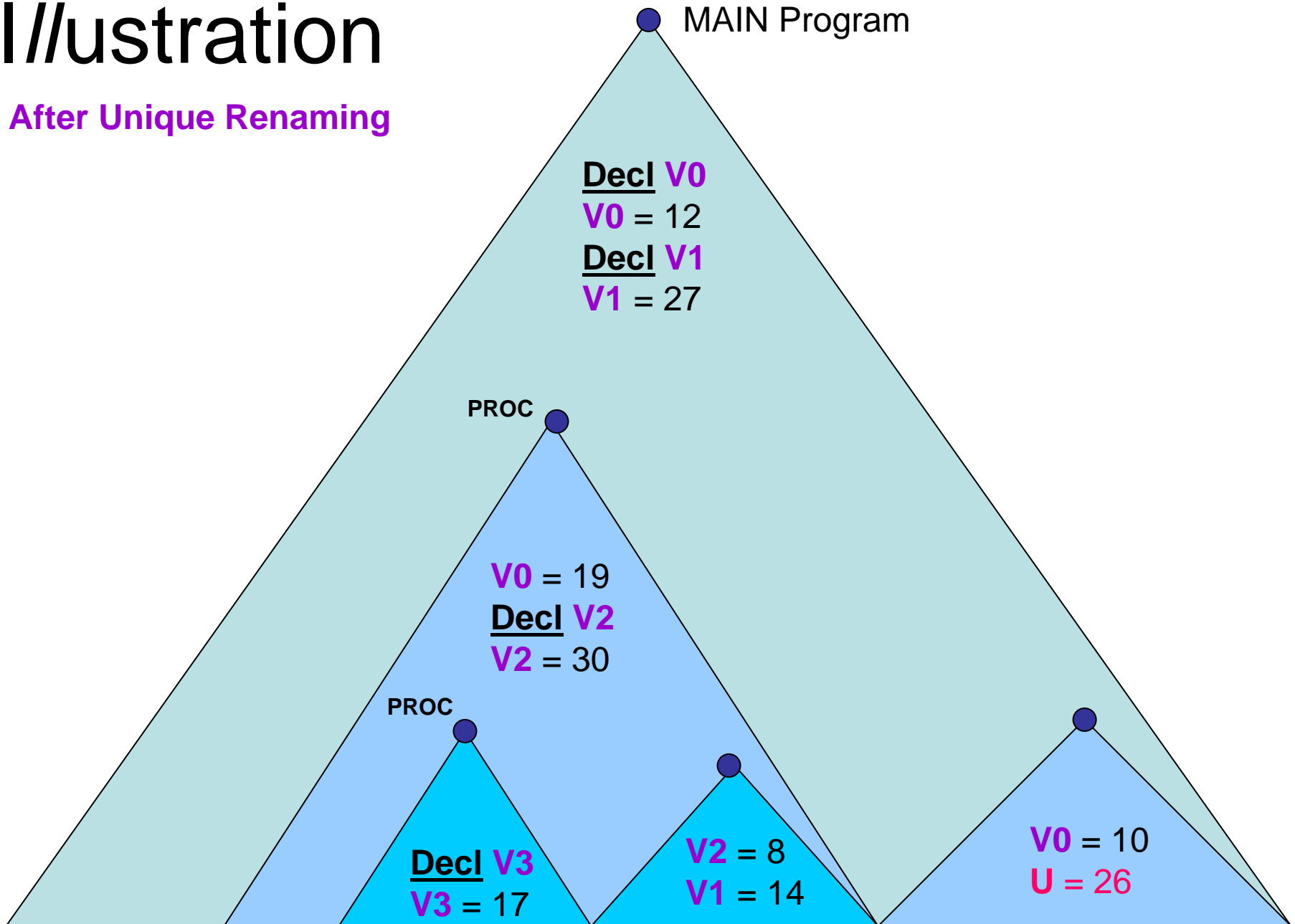
# Illustration

After Unique Renaming



# Illustration

After Unique Renaming





# Advice:

- In the Assessment (*Unlucky-Raven-Friday the 13<sup>th</sup> of April*) you must **show**:
  - the “**links**” from the AST nodes into the Table
  - the **Table-Entries** for all the AST nodes
- Type-Checking (num, string, bool) is NOT part of this Sub-Project 2a!
  - ➔ Type-Checking will come only later, in 2b.

And now :

***Happy Pair-Programming!***

