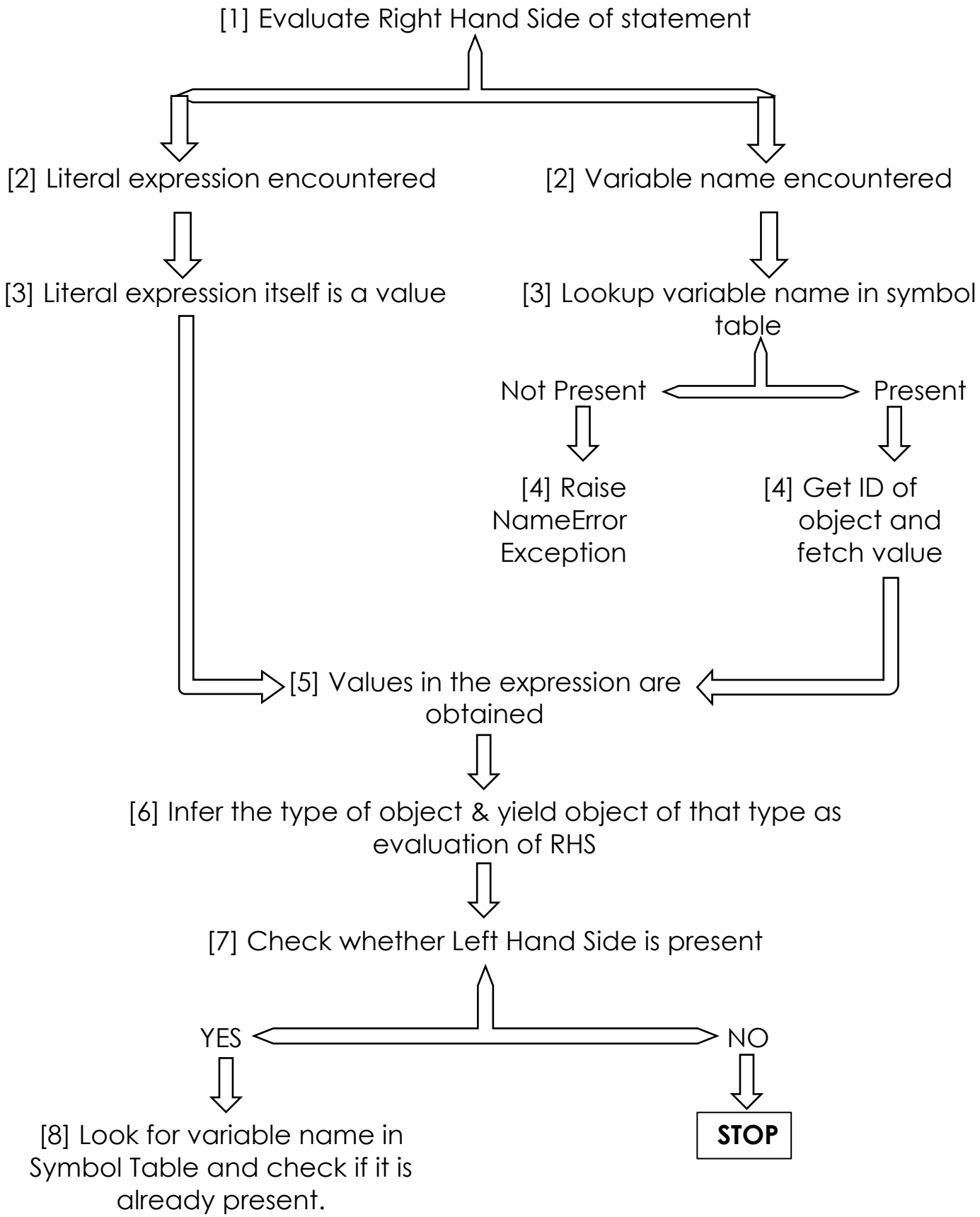
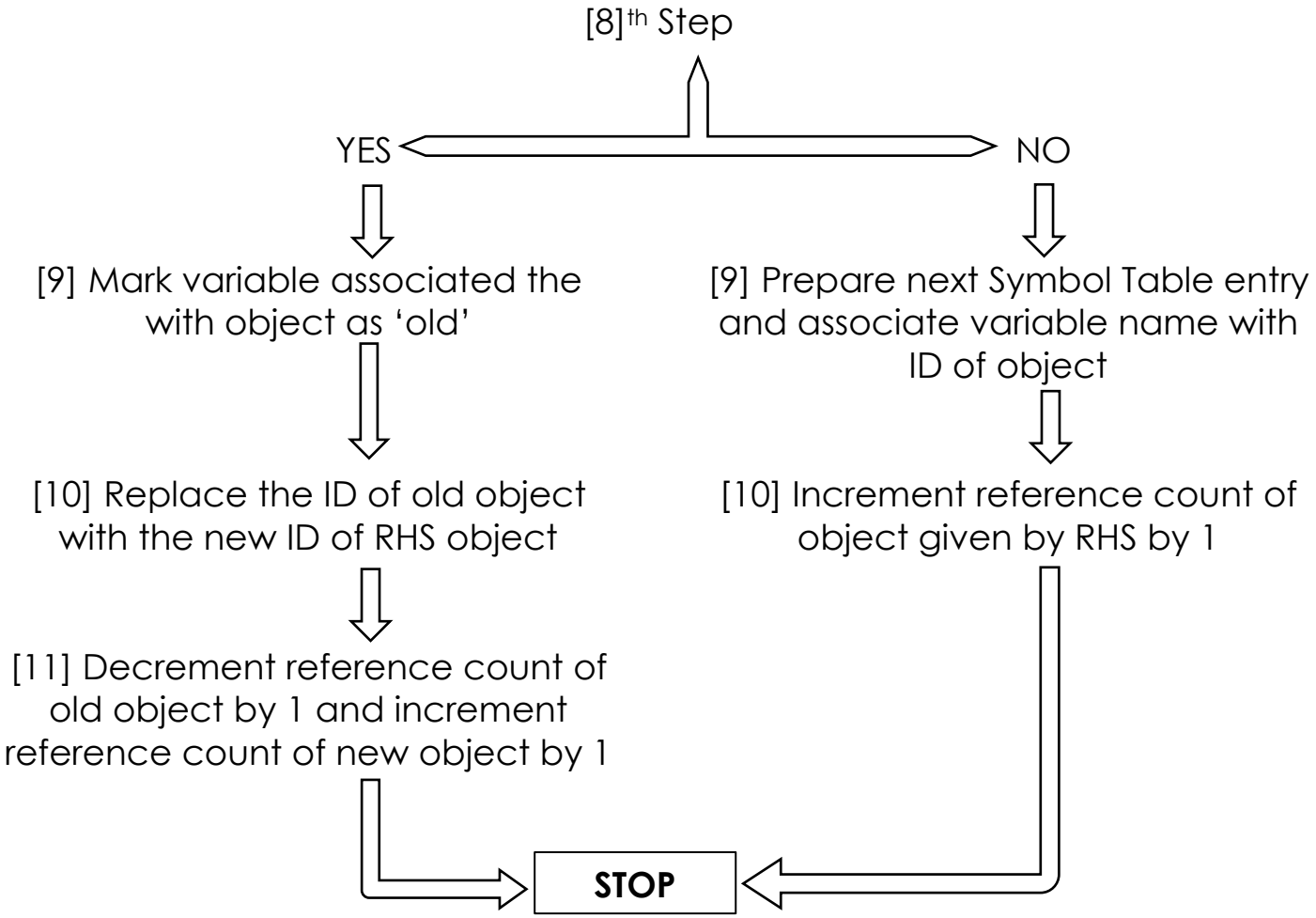
ASSIGNMENT STATEMENT ALGORITHM IN PYTHON

- **Diagram:**



Core Programming Academy – An academy by Yogeshwar Shukla
Mobile: +91 956 154 7043
Email: coreprogrammingacademy@gmail.com



- **General Syntax**:
  Left Hand Side = Right Hand Side
  where right hand side is an expression and left hand is a variable name.

- **Examples**:
  n1 = 10
  n2 = 20
  sum = n1 + n2
  s = "Hello"
  etc.

- **Generalized algorithm for evaluating any assignment statement in Python.**
    - **STEP I**: Evaluate right hand side expression.
        1. If literal expression is encountered, then that literal itself is a value.
        2. If literal expression is a variable name, then look up the variable name in the symbol table.
            1. If not found: raise NameError Exception.
            2. If found, then get the 'id' of an object associated with the variable name and fetch value component from it.
        3. Applying STEP I – 1 (in case of literal) and STEP I – 2 in case variable name, all the values in the expression are available now.
    - **STEP II:**
        1. Apply all the operators in the expressions on the obtained values in STEP – I and get the final value of an expression.
        2. Infer the type of the value obtained in STEP I – 4.
        3. Search the object with obtained value in STEP II – 1 and obtained type in STEP II – 2 in the object pool.
            1. If found, then get the 'id' of that object.
            2. If not found, then allocate object with the obtained value and obtained type and get it's 'id'
        4. Check if Left Hand Side variable name is present or not.
            1. If not, then STOP.
            2. If yes, then go to STEP III.
    - **STEP III:**
        1. Look up for the Left Hand Side variable in the symbol table. If not found, then execute STEP III – 2 and if found then execute STEP III – 3
        2. Handler of the case – "LHS variable name not found" –
            1. Create a new symbol table entry which binds obtained LHS variable name and 'id' of an object in STEP III – 3.
            2. Increment the reference count of an object by 1.
            3. STOP
        3. Handler of the case – "LHS variable already present"
            1. Get the 'id' of associated object in the symbol table and term it as an 'old id'. And let's term id obtained in STEP II -3 as a 'new id'
            2. In the existing symbol table entry of LHS variable name, replace 'old id' by 'new id'.
            3. Decrement the reference count of 'old id' by 1 and increment the reference count of 'new id' by 1.
            4. STOP.