# Scanning

1. Identify lexical tokens of a language. Start from an example:

for in n //python for in n

for(i=0; i<n; i++) //C

for ( i = 0 ; i < n ; i ++ )



VAR

H,F,K,A: INTEGER;

BEGIN

READ(H);

READ(Y);

K:=H+Y;

IF K>5 THEN BEGIN

A:=K+H

END

ELSE

A:=K+Y;

WRITE(A)

END.

## Symbol Table represented as a Sorted Table

|  |  |
| --- | --- |
| **Position** | **Token** |
| 1 | 5 |
| 2 | A |
| 3 | F |
| 4 | H |
| 5 | K |
| 6 | Y |
|  |  |

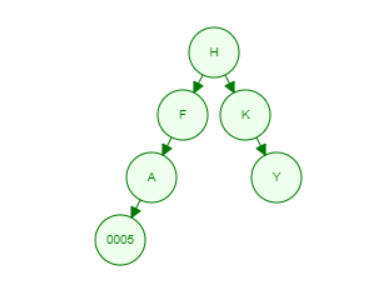
**Program Internal Form (PIF)**

|  |  |
| --- | --- |
| “ |  |
| 5 | -1 |
| 0 | 4 |
| 24 | -1 |
| 0 | 3 |
| 24 | -1 |
| 0 | 5 |
| 24 | -1 |
| 0 | 2 |
| 22 | -1 |
| 6 | -1 |
| 23 | -1 |
| 9 | -1 |
| 11 | -1 |
| 28 | -1 |
| 0 | 4 |
| 29 | -1 |
| 23 | -1 |
| 11 | -1 |
| 28 | -1 |
| 0 | 6 |
| 29 | -1 |
| 23 | -1 |
| 0 | 4 |
| 36 | -1 |
| 0 | 4 |
| 26 | -1 |
| 0 | 5 |
| 23 | -1 |
| 16 | -1 |
| 0 | 4 |
| 34 | -1 |
| 1 | 1 |
| 17 | -1 |
| 9 | -1 |
| 0 | 2 |
| 36 | -1 |
| 0 | 5 |
| 26 | -1 |
| 0 | 4 |
| 10 | -1 |
| 18 | -1 |
| 0 | 2 |
| 36 | -1 |
| 0 | 5 |
| 26 | -1 |
| 0 | 6 |
| 23 | -1 |
| 12 | -1 |
| 28 | -1 |
| 0 | 2 |
| 29 | -1 |
| 10 | -1 |
| 25 | -1 |

Symbol Table as a HashTable - hash function

|  |  |
| --- | --- |
| **Position** | **Token** |
| **...** |  |
| **12** | H |
| **...** |  |
| **...** |  |
| **19** | F A |
| **…** |  |
| **23** | K |
| **...** |  |
| **40** | Y |
| **41** | 5 |
| **...** |  |

## Symbol Table as a BST



|  |  |  |  |
| --- | --- | --- | --- |
| **Position** | Token | Father | Sibling |
| **1** | H | -1 | -1 |
| **2** | F | 1 | 3 |
| **3** | K | 1 | 2 |
| **4** | A | 2 | -1 |
| **5** | Y | 3 | -1 |
| **6** | 5 | 4 | -1 |

## Lexical errors:

VAR

H,2F,K,A: INTEGER;

BEGIN

READY(H);

‘KH’

‘#’

READ(Y);

K:=H+Y;

IF K>5 THEN BEGIN

A:=K+H

END

ELSE

A:=K+Y;

WRITE(A)

END.