ASSIGNMENT NO: 3

Title :- Generate & Populate appropriate Symbol Table.

Problem Statement:-

Design suitable data structures to generate & populate appropriate symbol table.

Objective :-

Understand the symbol table is an important data structure created & maintained by compilers.

S/w packages & hardware apparatus :-
- 64-bit open source Linux.
- Eclipse IDE.
- Java

Outcome :-

The students will be able to
- Generate Symbol Table using suitable data structures
- Study of Symbol Table.

Theory :-

A symbol table is a data structure used by the compiler, where each identifier in program's source code is stored along with information associated with it relating to its declaration. It stores identifier as well as it's associated attributes like scope, type, line, number of occurence, etc.

Symbol table can be implemented using various data structures like:

- Linked list.
- Hash Table.
- Tree.
- Binary Search Tree.

A common data structure used to implement a symbol table is Hash Table.

Implementation of Symbol Table:

Following are commonly used data structure for implementing symbol table:-

1. List –

- In this method an array is used to store names and associated information.
- A pointer "available" is maintained at the end of all stored records & new names are added in the order as they arrive.
- To search for a name we start from beginning of list till available pointer & if not found we get an error "use of undeclared name".
- While insertig a new name we must ensure that it is not already present otherwise error occurs i.e. "Multiple defined name".
- Insertion is fast O(1), but lookup is slow for large tables.

Page No.
Date

O(n) is average.

- Advantage is that it takes minimum amount of space.

2. Linked List :-

- This implementation is using linked list. A link field is added to each record.
- Searching of names is done in order pointed by link of link field.
- A pointer "First" is maintained to point to first record of symbol table.
- Insertion is fast O(1), but lookup is slow for large tables - O(n) on average.

3. Hash Table :-

- In hashing scheme two tables are maintained - a hash table & symbol table and is the most commonly used method to implement symbol tables.
- A hash table is an array with index range: 0 to tablesize - 1. These entries are pointer pointing to names of symbol table.
- To search for a name we use hash function that will result in any integer between 0 to tablesize -1.
- Insertion & lookup can be made very fast - O(1).
- Advantage is quick search is possible and disadvantage is that hashing is complicated to implement.

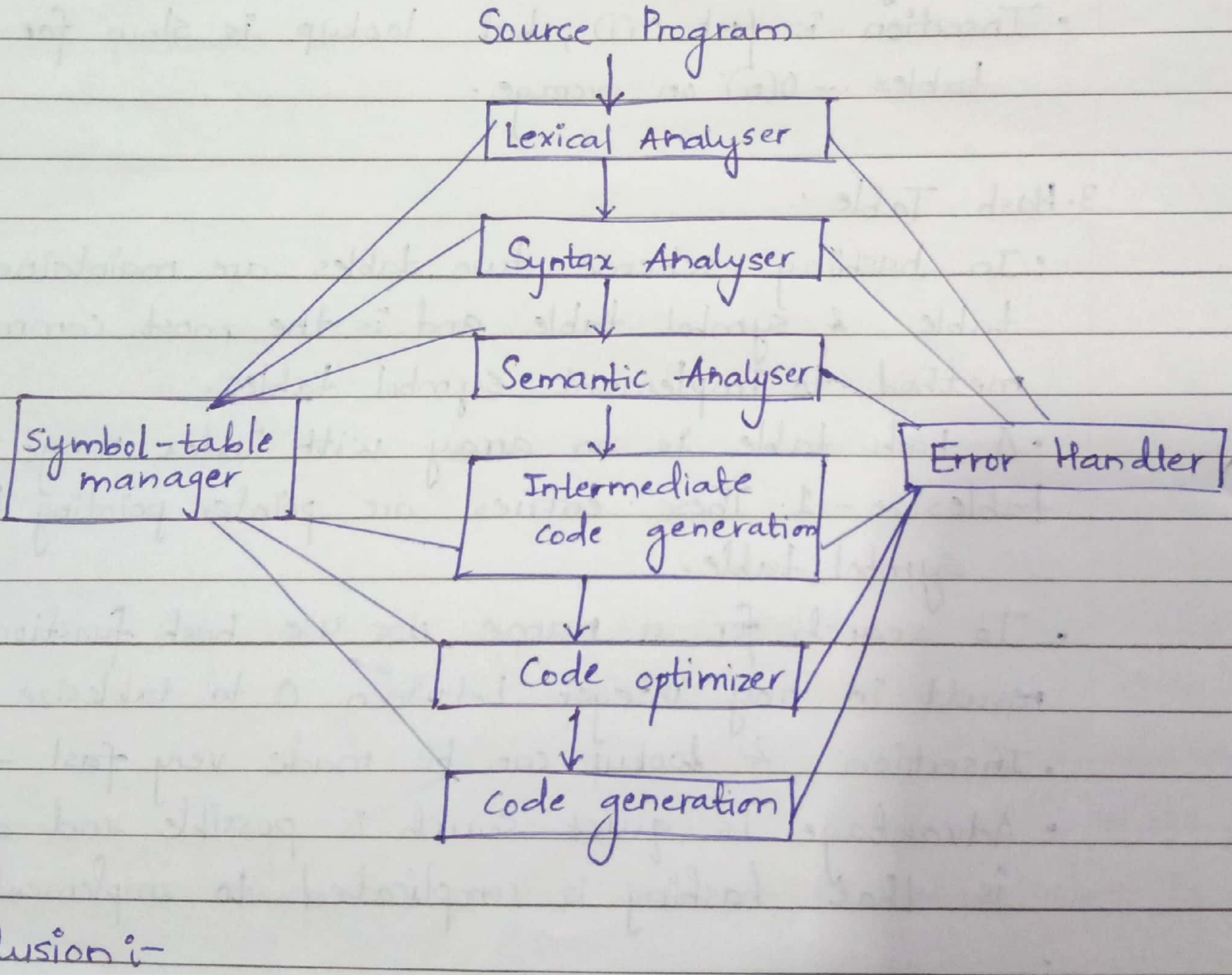
Scanned by CamScanner

Page No.
Date

Binary Search Tree :

- Another approach to implement symbol table is to use binary search tree i.e. we add two link fields i.e. left & right child.
- All names are created as child of root node that always follow the property of binary search tree.
- Insertion & lookup are $O(\log_2 n)$ on average.

PHASES OF COMPILER .



Conclusion :-

Hence, designed suitable data structures to generate & populate appropriate symbol Table.