symbol table implementation using Hash Table.

Symbol table is an important data structure created and maintained by compilers in order to store information about the occurrence of various identifiers such as variable names, function names, objects, classes, interfaces, etc. Symbol table is used by both the analysis and the synthesis parts of a compiler. Symbol table can be implemented in one of the following ways:

- Linear (sorted or unsorted) list
- Binary Search Tree
- Hash table
- And other ways.

IMPLEMENTATION OF OF SYMBOL TABLE USING HASH TABLE

Symbol Table : The compiler creates and maintains a data structure to store information about the occurrence of various entities
such as variable and function names, objects and classes, etc.
This kind of data structure is known as a "Symbol table".

Items stored in symbol table:

- Variable names and constants
- Procedure and function names
- Literal constants and strings
- Compiler generated temporaries
- Labels in source languages

In Hash Table:
Hash table technique is suitable for searching and hence it is implemented in compiler.
Among all, symbol tables are mostly implemented as hash tables, where the source code symbol itself is treated as a key for the hash function and the return value is the information about the symbol.
In hashing scheme, two tables are maintained – a hash table and symbol table and are the most commonly used method to implement symbol tables.
A hash table is an array with an index range: 0 to table size – 1. These entries are pointers pointing to the names of the symbol table.
To search for a name we use a hash function that will result in an integer between 0 to table size – 1.
Insertion and lookup can be made very fast – O(1).
The advantage is quick to search is possible and the disadvantage is that hashing is complicated to implement.

Insertion Time –
As in a hash table insertion takes constant time so the time required for insertion is O(1).
Lookup Time –
Searching in a hash table requires constant time so the time required is O(1).

Advantages of using Hashing for symbol table implementations : It is very efficient.

Disadvantages of using Hashing for symbol table implementations :
The disadvantage of this implementation is when there are too many collisions the time complexity increases to O(n).