

CSE 310

Assignment 01

Construction of a Symbol-Table

The target of this sessional is to implement a functional compiler. As the first step towards the development of an interpreter, we will build a symbol-table in this assignment. At this initial stage of the project, we will omit many details regarding an actual symbol-table and we will simply adhere to the basic concept that “a symbol-table is an efficient data-dictionary for the symbols used in a program”. Thus, our focus in this assignment is to construct a simple hash-based data-dictionary based on chaining.

Input:

The input to your program will be a sequence of two-tuples, where each element in each tuple is a string. An example of input sequence is given below.

- int, INTEGER
- myFunction, FUNCTION
- x, IDENTIFIER
- 5, NUMBER

The first element of each tuple will be the key of the record to be stored in the symboltable. Hence, you have to apply the hash function on the first element of each tuple.

Implementation Issues:

Implement the following two classes:

- class SymbolInfo: The definition of this class will grow gradually throughout the development of this project. For this assignment, we simply need two members, one for storing the symbol (e.g. “x”) and another for storing the type of the symbol (e.g. “IDENTIFIER”).

- class SymbolTable: Since our symbol-table will be a hash-table based on chaining, we will have to start with an array of pointers where each pointer points to a list of nodes of type class SymbolInfo. class SymbolTable will have such an array of pointers. For this assignment, the choice of the size of this array, as well as of the hash function is left upto you(20 can be preferable). In addition to this array of pointers, class SymbolTable will have three functions for the following purposes.
- insert(): to insert a new symbol along with its type into the symbol table.
- lookup(): to lookup whether a given symbol already exists in the symbol table or not
- dump(): to dump the contents of the symbol table to the console.

Some Other issues:

- Always avoid hard-coding and try to use dynamic memory allocation in this project.
- You have to do this whole project in Linux platform (Ubuntu is preferable, in lab 12.4 (precise) is available)
- To compile a cpp code in linux
 - gcc & g++ must be installed in machine
 - Using the following command you can compile a cpp code suppose named sample.cpp


```
g++ -o temp sample.cpp
```

```
./temp
```

Submission: In next sessional class

Prepared By
Shampa Shahriyar
Lecturer,CSE-BUET