United International University (UIU)

Dept. of Computer Science & Engineering (CSE)

ASSIGNMENT 1 OFFLINE

Course Code: CSI 412 Course Title: Compiler
Section: SB

Construction of a Rudimentary Symbol-Table

1 Introduction

In this course, we will try to develop a complete compiler, although our source language may vary from the one described there. As the first step towards the development of a compiler, 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.

2 Inputs

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, ID
- 5, NUM

The first element of each tuple will be the name of the record to be stored in the symbol table. Hence, you have to apply the hash function on the first element of each tuple that is the name.

3 Implementation Issues

Implement the following two classes:

1. 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.

In addition to this array of pointers, class SymbolTable will have three functions for the following purposes.

insert(): to insert a new symbol /name along with its type into the symbol table,

search(): to search a new symbol /name along with its type from the symbol table,

delete(): to delete a symbol /name along with its type into the symbol table

show(): to show the contents of the symbol table to the console.

Hash Function: You can for now take the sum of the ASCII value and mod it with any prime number say 53 to determine the hash function.

4 Deadlines

• Deadline to submit the assignment: 27th October, 2015.

References

[AHU02] A. V. Aho, R. Sethi and J. D. Ullman, Compilers: Principles, Techniques, and Tools, Addison-Wesley, 2002.