# Fr. Conceicao Rodrigues College of Engineering Department of Computer Engineering

**Academic Term : Jan-May 2024 - 25**

**Class : T.E. (Computer - A)**

**Subject Name : System Programming and Compiler Construction**

**Subject Code : (CPC601)**

|  |  |
| --- | --- |
| **Practical No:** | 1 |
| **Title:** | To write a program for implementing Symbol Table. |
| **Date of Performance:** | 4/02/2025 |
| **Date of Submission:** | 10/02/2025 |
| **Roll No:** | 9913 |
| **Name of the Student:** | Mark Lopes |

**Evaluation:**

|  |  |  |
| --- | --- | --- |
| **Sr. No** | **Rubric** | **Grade** |
| **1** | **Time Line (2)** |  |
| **2** | **Output(3)** |  |
| **3** | **Code optimization (2)** |  |
| **4** | **Postlab (3)** |  |

**Signature of the Teacher :**

**Experiment No 1**

## AIM:

To write a program for implementing Symbol Table.

## ALGORITHM

**Step1:** Start the program for performing insert, display, delete, search and modify option in symbol table

**Step2:** Define the structure of the Symbol Table

**Step3:** Enter the choice for performing the operations in the symbol Table

**Step4:** If the entered choice is 1, search the symbol table for the symbol to be inserted. If the symbol is

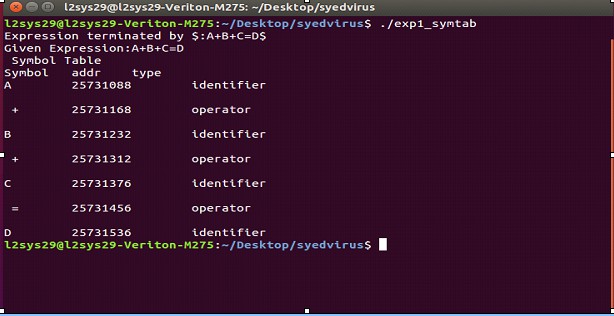
already present, it displays “Duplicate Symbol”. Else, insert the symbol and the corresponding address in

the symbol table.

**Step5:** If the entered choice is 2, the symbols present in the symbol table are displayed. **Step6:** If the entered choice is 3, the symbol to be deleted is searched in the symbol table. **Step7:** If it is not found in the symbol table it displays “Label Not found”. Else, the symbol is deleted.

**Step8:** If the entered choice is 5, the symbol to be modified is searched in the symbol table.

## Sample Input and Output:

****

**#include<stdio.h>**

**#include<string.h>**

**#include<stdlib.h>**

**#include<ctype.h>**

**#define MAX 100**

**typedef struct{**

**char symbol;**

**char type[20];**

**} Symbol;**

**Symbol symbolTable[MAX];**

**int symbolCount = 0;**

**int search(char symbol)**

**{**

**for (int i = 0; i < symbolCount; i++)**

**{**

**if (symbolTable[i].symbol == symbol){**

**return i; // found**

**}**

**}**

**return -1; // not found**

**}**

**void insert(char symbol, char \*type)**

**{**

**if (search(symbol) != -1)**

**{**

**printf("Duplicate Symbol: %c \n", symbol);**

**return;**

**}**

**symbolTable[symbolCount].symbol = symbol;**

**strcpy(symbolTable[symbolCount].type, type);**

**symbolCount++;**

**}**

**void createSymbolTable(char \*expr)**

**{**

**printf("The expression is: %s \n", expr);**

**for (int i = 0; expr[i] != '\0'; i++)**

**{**

**if (expr[i] == '$')**

**{**

**break; // expression terminated**

**}**

**if (isalpha(expr[i]))  // If the character is an alphabet (identifier)**

**{**

**insert(expr[i], "identifier");**

**}**

**else if (expr[i] == '+' || expr[i] == '-' || expr[i] == '=' || expr[i] == '\*')  // If it's an operator**

**{**

**insert(expr[i], "operator");**

**}**

**else if (isdigit(expr[i]))  // If it's a digit (constant)**

**{**

**insert(expr[i], "constant");**

**}**

**}**

**printf("Table is created.");**

**}**

**void display\_table()**

**{**

**printf("\nSymbol Table\n");**

**printf("Symbol      Type\n");**

**for(int i  = 0; i < symbolCount; i++)**

**{**

**printf("%c      %s\n", symbolTable[i].symbol, symbolTable[i].type);**

**}**

**}**

**void delete\_Symbol(char symbol)**

**{**

**int pos = search(symbol);**

**if(pos == -1){**

**printf("Symbol not found.");**

**return;**

**}**

**for(int i = pos; i < symbolCount -1; i++)**

**{**

**symbolTable[i] = symbolTable[i+1];**

**}**

**symbolCount--;**

**printf("Symbol %c is deleted successfully.\n", symbol);**

**}**

**int main()**

**{**

**char expr[MAX];**

**int choice;**

**while(1) {**

**printf("Enter your choice: \n1. Enter your expression\n2. Create Table\n3. Display Table\n4. Delete Symbol\n5.Exit\n");**

**scanf("%d", &choice);**

**// Clear the input buffer**

**while(getchar() != '\n');**

**switch (choice)**

**{**

**case 1:**

**printf("Enter your expression: ");**

**fgets(expr, MAX, stdin);  // Read the expression**

**// Remove newline character if present**

**expr[strcspn(expr, "\n")] = 0;**

**break;**

**case 2:**

**createSymbolTable(expr);**

**break;**

**case 3:**

**display\_table();**

**break;**

**case 4:**

**char n;**

**printf("Enter the symbol to be deleted");**

**scanf("%c",&n);**

**delete\_Symbol(n);**

**break;**

**case 5:**

**printf("Exiting program.....");**

**return;**

**default:**

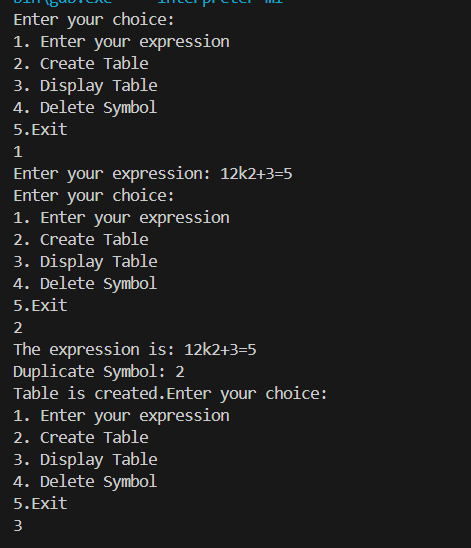
**printf("Invalid choice. Please enter a valid option.\n");**

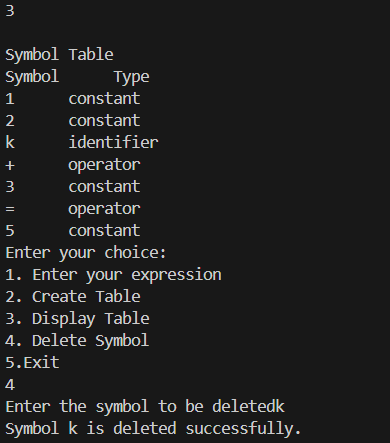
**break;**

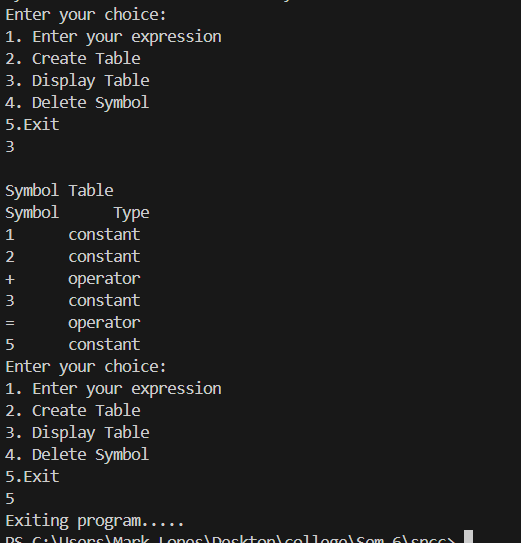
**}**

**}**

**}**

****

****

****

**Postlab Questions:**

1. Explain different phases of compiler. Illustrate all the output after each phase for the following statement

a= b +c -d \*5

