Bangladesh University of Business & Technology

Course Code : CSE 324

Course Title : Compiler Design Lab

Lab Report : 02

Submitted By :

MD. Rakibul Hasan

Id : 17182103245

Section : 06

Intake : 38

Submitted To :

Adeeba Anis

Lecturer

Department of CSE

BUBT

Lab No: 02

Problem name: A C program to identify valid operator and keywords.

Problem Statement: In this problem, we need to identify the valid operator and keyword. So, for that we have to know what actually a keyword and operator is.

In a programming language, a keyword is also known as reserved word. We can't use it as a variable or constant name. In c language there are 32 keywords. On the other hand, an operator is a symbol that tells the compiler to perform specific mathematical or logical functions. Some well-known keywords and operators are:

- Keywords: int, float, double, break, while, const, continue, char, string etc.
- Operators: ==, !=, >=, <= etc.

This problem is solved with c language. At first, I'm going to declare the header file and then I'll create the main function. Inside the main method, I'm going to take a string input. And then I'll check that, if the given string is a keyword. If it is, then the program will print, "This is a keyword."

Else, if the given string is operator, then It'll print, "This is an operator."

Else, if the inputted string if not belongs to a keyword or an operator, then It will print, "This is not a keyword nor an operator."

Code:

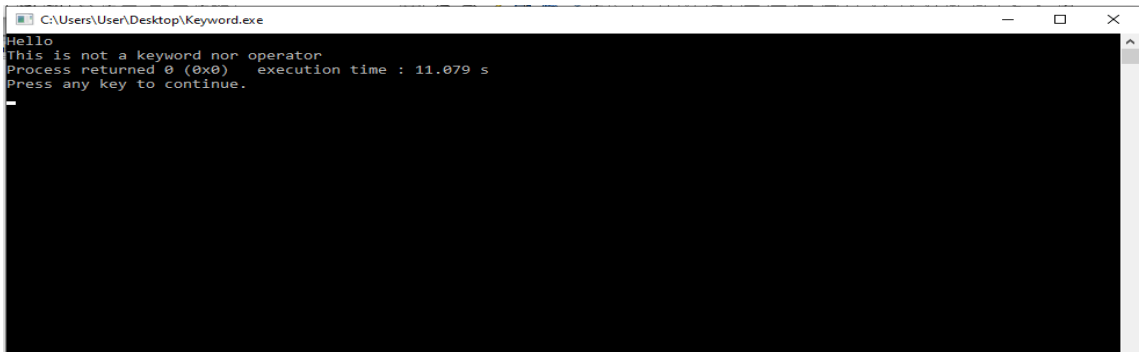
```
#include <stdbool.h>
#include <stdio.h>
#include <string.h>
bool Keyword(char* str)
{
   if (!strcmp(str, "auto") || !strcmp(str, "default")
      || !strcmp(str, "signed") || !strcmp(str, "enum")
      ||!strcmp(str, "extern") || !strcmp(str, "for")
      || !strcmp(str, "register") || !strcmp(str, "if")
      || !strcmp(str, "else")  || !strcmp(str, "int")
      || !strcmp(str, "while") || !strcmp(str, "do")
      || !strcmp(str, "break") || !strcmp(str, "continue")
      || !strcmp(str, "double") || !strcmp(str, "float")
      || !strcmp(str, "return") || !strcmp(str, "char")
      || !strcmp(str, "case") || !strcmp(str, "const")
      || !strcmp(str, "sizeof") || !strcmp(str, "long")
      || !strcmp(str, "short") || !strcmp(str, "typedef")
      || !strcmp(str, "switch") || !strcmp(str, "unsigned")
      || !strcmp(str, "void") || !strcmp(str, "static")
      || !strcmp(str, "struct") || !strcmp(str, "goto")
      || !strcmp(str, "union") || !strcmp(str, "volatile"))
      return (true);
   return (false);
}
```

```
int operator(char* str)
{
   if(!strcmp(str, "="))
      printf("This is equal operator");
   else if(!strcmp(str, "!="))
      printf("This is not equal operator");
   else if(!strcmp(str, "+"))
      printf("This is plus operator");
   else if(!strcmp(str, "-"))
   {
      printf("This is minus operator");
   }
   else if(!strcmp(str, "*"))
   {
      printf("This is multiple operator");
   }
   else if(!strcmp(str, "/")){
      printf("This is divide operator ");
   }
   else
      printf("This is not a keyword nor operator");
return 0;
}
```

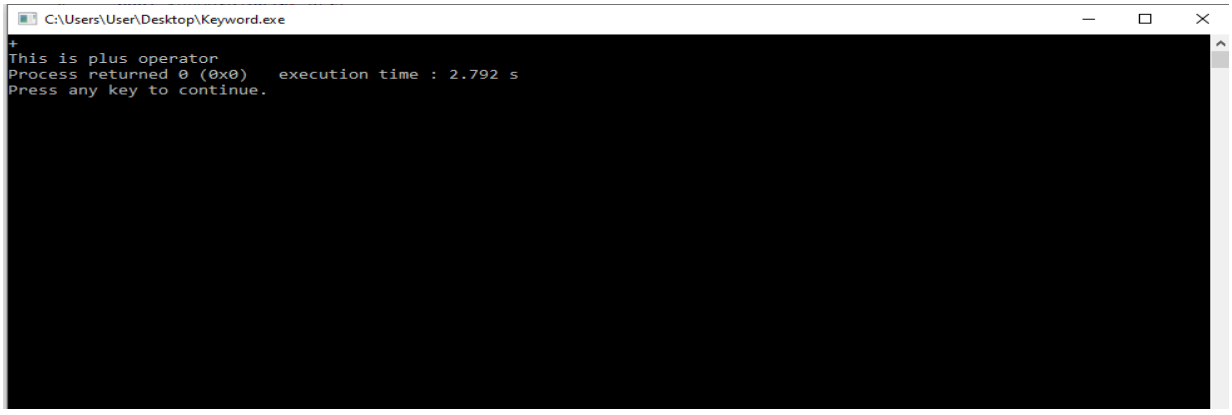
```
int main()
{
   char ch[100];
   scanf("%s",&ch);
   if(Keyword(ch))
   {
      printf("This is a keyword");
   }
   else
   {
      operator(ch);
   }
   return 0;
}
```

Sample input & output :

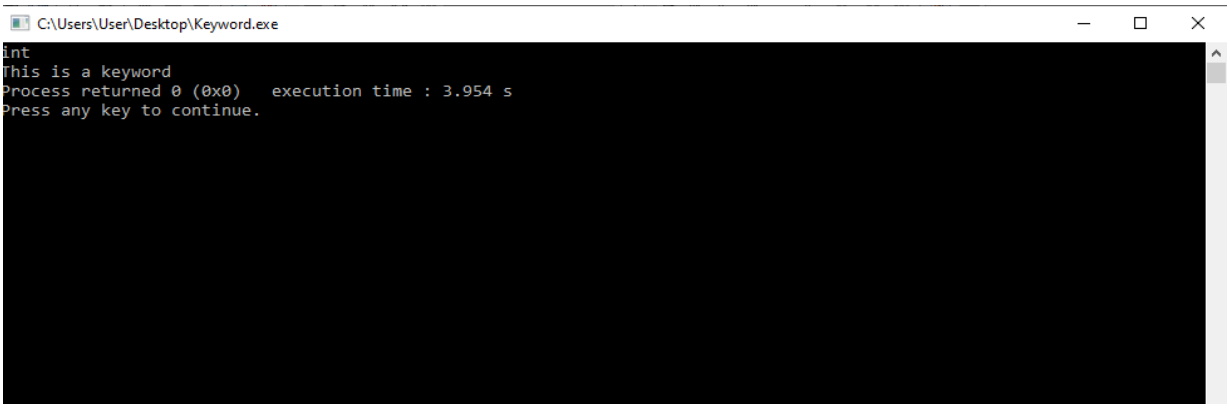
Test case 1:



Test case 2:



Test case 3:


C:\Users\User\Desktop\Keyword.exe
int
This is a keyword
Process returned 0 (0x0)   execution time : 3.954 s
Press any key to continue.