

Ex. No.: 1	Date: 31-12-2025
Title: Identify and Classify Identifiers Using Transition Diagram	
Aim: Implement transition diagram for identifying an identifier and classify whether it is either variable or array or function or structure.	
Algorithm:	
<p>Step 1: Start</p> <p>Step 2: Read the input string.</p> <p>Step 3: Check the first character. If it is not a letter or _, print Invalid input and stop.</p> <p>Step 4: Check remaining characters one by one. If any character is not a letter, digit, _, [], (or), print Invalid input and stop.</p> <p>Step 5: Find the length of the input string.</p> <p>Step 6: If the last two characters are [], print Valid identifier: ARRAY.</p> <p>Step 7: Else if the last two characters are (), print Valid identifier: FUNCTION.</p> <p>Step 8: Else print Valid identifier: VARIABLE.</p> <p>Step 9: Stop.</p>	
Program:	
<pre>#include <stdio.h> #include <ctype.h> #include <string.h> int isValidIdentifier(char str[]) { if (!(isalpha(str[0]) str[0] == '_')) return 0; for (int i = 1; str[i] != '\0'; i++) { if (!(isalnum(str[i]) str[i] == '_' str[i] == '[' str[i] == ']' str[i] == '(' str[i] == ')')) } }</pre>	

```
        return 0;      }

    return 1;

}

int main() {

    char input[50];

    int len;

    printf("Enter an identifier: ");

    scanf("%s", input);

    len = strlen(input);

    if (!isValidIdentifier(input)) {

        printf("Invalid input\n");

        return 0;

    }

    if (len > 2 && input[len - 2] == '[' && input[len - 1] == ']') {

        printf("Valid identifier: ARRAY\n");

    }

    /* Check for function */

    else if (len > 2 && input[len - 2] == '(' && input[len - 1] == ')') {

        printf("Valid identifier: FUNCTION\n");

    }

}
```

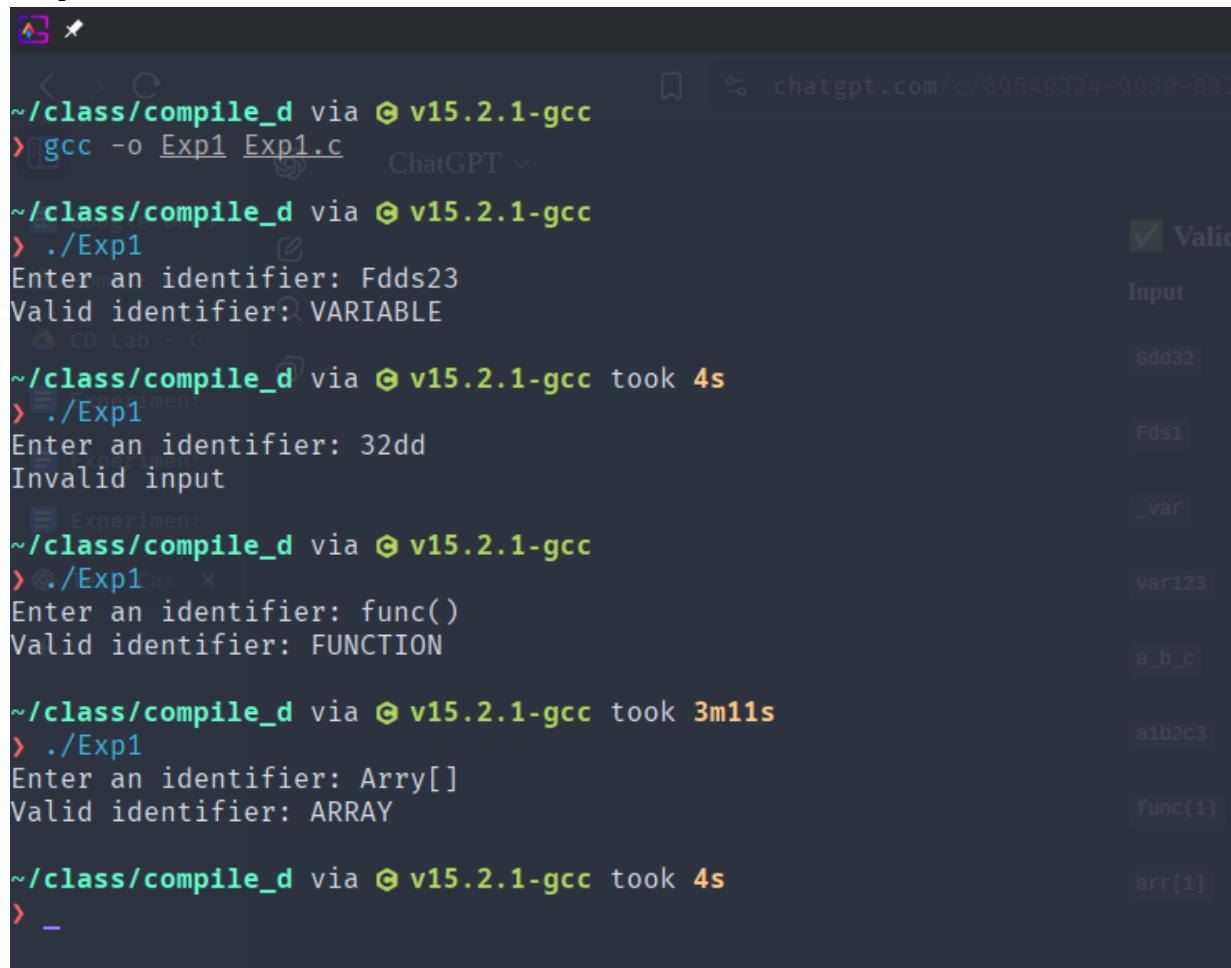
```
else { printf("Valid identifier: VARIABLE\n");  
}  
  
return 0;  
}
```

Result:

Program executed successfully and verified output.

Output:

Output Screenshot



The screenshot shows a terminal window with the following session:

```
~/class/compile_d via v15.2.1-gcc  
gcc -o Exp1 Exp1.c  
~/class/compile_d via v15.2.1-gcc  
./Exp1  
Enter an identifier: Fdds23  
Valid identifier: VARIABLE  
~/class/compile_d via v15.2.1-gcc took 4s  
./Exp1  
Enter an identifier: 32dd  
Invalid input  
~/class/compile_d via v15.2.1-gcc  
./Exp1  
Enter an identifier: func()  
Valid identifier: FUNCTION  
~/class/compile_d via v15.2.1-gcc took 3m11s  
./Exp1  
Enter an identifier: Arry[]  
Valid identifier: ARRAY  
~/class/compile_d via v15.2.1-gcc took 4s  
_
```

The terminal shows the execution of a C program named `Exp1.c` using `gcc`. It prompts the user to enter an identifier and checks if it is valid. The identifiers `Fdds23`, `32dd`, `func()`, `Arry[]`, and `_` are checked, with `Fdds23` being valid and the others invalid. The program also prints the time taken for execution.