

Experiment - 6

Aim: To Design predictive parser for the given language.

Program:

```
#include <stdio.h>
#include <string.h>

char input[20];
int len, ln, err = 0;

void E();
void E1();
void T();
void T1();
void F();
void match(char topChar);

void E() {
    T();
    E1();
}

void E1() {
    if (*input == '+') {
        match('+');
        T();
        E1();
    } else {
        return;
    }
}

void T() {
    F();
    T1();
}

void T1() {
    if (*input == '*') {
        match('*');
        F();
        T1();
    } else {
        return;
    }
}

void F() {
    if (*input == '(') {
```

Exp No: 6
Name: Pavan Kumar Ch

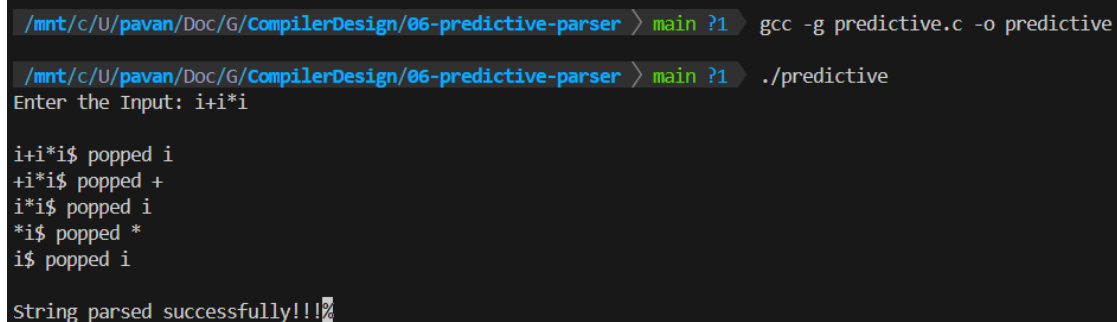
Date:
Reged.no: 22501A05E0

```
        match('(');
        E();
        match(')');
    } else {
        match('i');
    }
}

void match(char topChar) {
    if (*input == topChar) {
        printf("\n%s popped %c", input, topChar);
        ln++;
        strcpy(input, &input[1]);
    } else {
        printf("\nError %c didn't produced by any production at this place", *input);
        err++;
    }
}

int main() {
    printf("Enter the Input: ");
    gets(input);
    len = strlen(input);
    input[len] = '$';
    input[len + 1] = '\0';
    E();
    if (err == 0 && ln == len) {
        printf("\n\nString parsed successfully!!!");
    } else {
        printf("\n\nString is not parsed successfully\nErrors occurred or Input contains invalid characters\n\n");
    }
    return 0;
}
```

Output:



```
/mnt/c/0/pavan/Doc/G/CompilerDesign/06-predictive-parser > main ?1 gcc -g predictive.c -o predictive
/mnt/c/0/pavan/Doc/G/CompilerDesign/06-predictive-parser > main ?1 ./predictive
Enter the Input: i+i*i

i+i*i$ popped i
+i*i$ popped +
i*i$ popped i
*i$ popped *
i$ popped i

String parsed successfully!!!
```

Result: Developed a predictive parser in C for the given language