CSA 1447-COMPILER DESIGN FOR SYNTAX SMITH

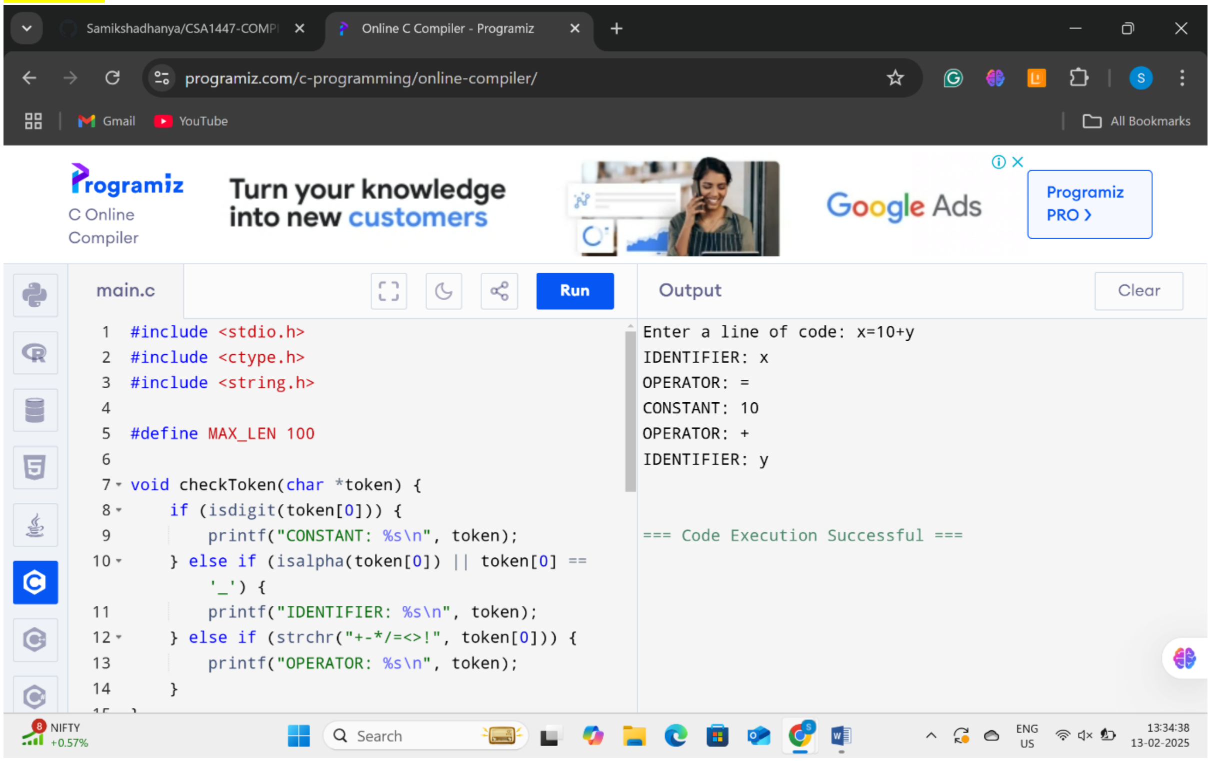
PRACTICAL PROGRAMS 1 - 7

EXPERIMENT-1

PROGRAM:

```
#include <stdio.h>

#include <ctype.h>

#include <string.h>

#define MAX_LEN 100

void checkToken(char *token) {

  if (isdigit(token[0])) {

    printf("CONSTANT: %s\n", token);

  } else if (isalpha(token[0]) || token[0] == '_') {

    printf("IDENTIFIER: %s\n", token);

  } else if (strchr("+-*/=<>!", token[0])) {

    printf("OPERATOR: %s\n", token);

  }

}

int main() {

  char input[MAX_LEN], buffer[MAX_LEN];

  int i = 0;

  printf("Enter a line of code: ");

  fgets(input, MAX_LEN, stdin);

  for (int j = 0; input[j] != '\0'; j++) {

    if (isspace(input[j])) {

      if (i != 0) {

        buffer[i] = '\0';

        checkToken(buffer);

        i = 0;

      }

    } else if (strchr("+-*/=<>!()", input[j])) {

      if (i != 0) {
            buffer[i] = '\0';
            checkToken(buffer);
            i = 0;
        }
        printf("OPERATOR: %c\n", input[j]);
    } else {
        buffer[i++] = input[j];
    }}
  return 0;
}
```

OUTPUT:



Samikshadhanya/CSA1447-COMP
Online C Compiler - Programiz
programiz.com/c-programming/online-compiler/
Gmail
YouTube
All Bookmarks
Programiz
C Online Compiler
Turn your knowledge into new customers
Google Ads
Programiz PRO
main.c
Run
Output
Clear
NIFTY +0.57%
Search
ENG US
13:34:38
13-02-2025

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

#define MAX_LEN 100

void checkToken(char *token) {
    if (isdigit(token[0])) {
        printf("CONSTANT: %s\n", token);
    } else if (isalpha(token[0]) || token[0] ==
        '_') {
        printf("IDENTIFIER: %s\n", token);
    } else if (strchr("+-*/=<>!", token[0])) {
        printf("OPERATOR: %s\n", token);
    }
```

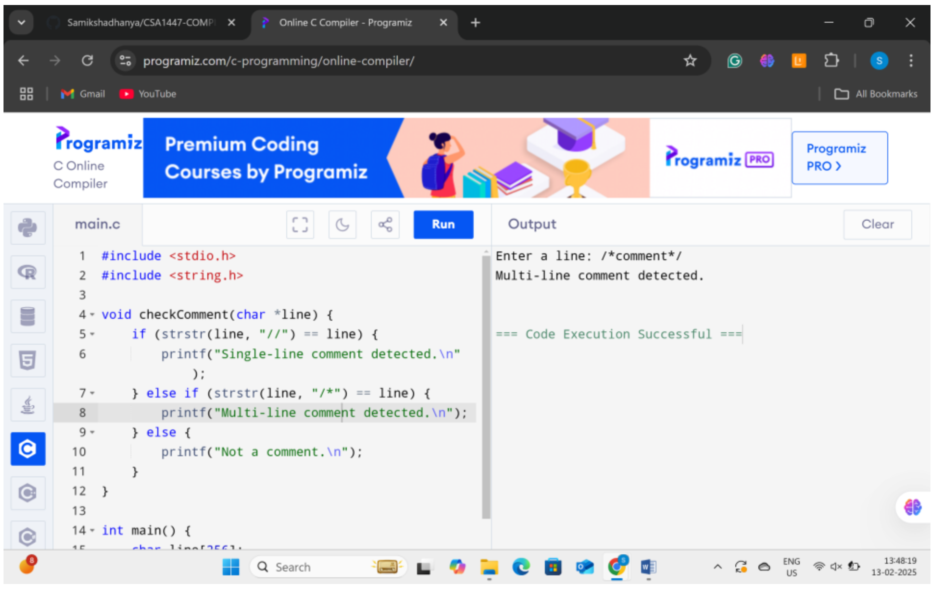
```
Enter a line of code: x=10+y
IDENTIFIER: x
OPERATOR: =
CONSTANT: 10
OPERATOR: +
IDENTIFIER: y

=== Code Execution Successful ===
```

EXPERIMENT-2

PROGRAM:

```
#include <stdio.h>
#include <string.h>
void checkComment(char *line) {
  if (strstr(line, "//") == line) {
      printf("Single-line comment detected.\n");
  } else if (strstr(line, "/*") == line) {
      printf("Multi-line comment detected.\n");
  } else {
      printf("Not a comment.\n");
  }
}
int main() {
  char line[256];
  printf("Enter a line: ");
  fgets(line, sizeof(line), stdin);
  checkComment(line);
  return 0;
}
```

OUTPUT:


Samikshadhanya/CSA1447-COMP
Online C Compiler - Programiz
programiz.com/c-programming/online-compiler/
Gmail
YouTube
All Bookmarks
Programiz C Online Compiler
Premium Coding Courses by Programiz
Programiz PRO
main.c
Run
Output
Clear
Search
ENG US
13:48:19 13-02-2025

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

void checkComment(char *line) {
    if (strstr(line, "//") == line) {
        printf("Single-line comment detected.\n"
            );
    } else if (strstr(line, "/*") == line) {
        printf("Multi-line comment detected.\n");
    } else {
        printf("Not a comment.\n");
    }
}

int main() {
```

```
Enter a line: /*comment*/
Multi-line comment detected.

=== Code Execution Successful ===
```

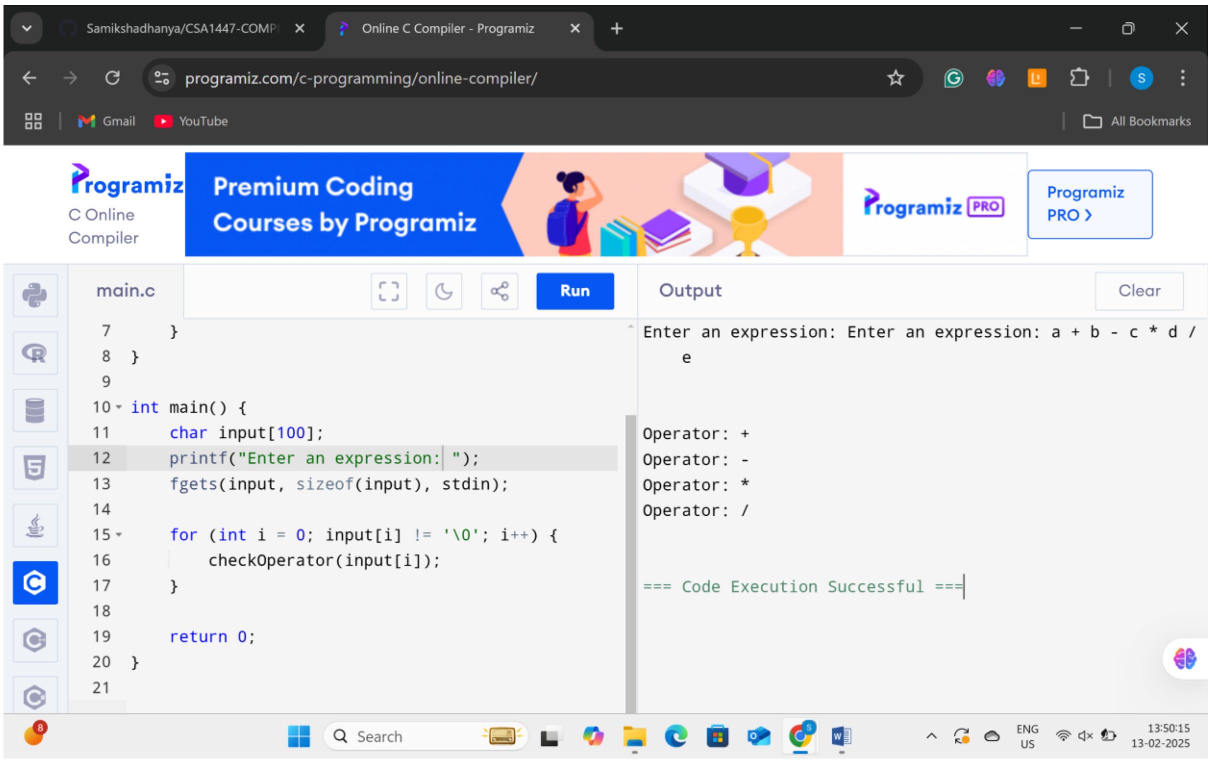
EXPERIMENT-3

PROGRAM:

```
#include <stdio.h>

#include <string.h>
```

```
void checkOperator(char ch) {
    if (ch == '+' || ch == '-' || ch == '*' || ch == '/') {
        printf("Operator: %c\n", ch);
    }
}

int main() {
    char input[100];
    printf("Enter an expression: ");
    fgets(input, sizeof(input), stdin);

    for (int i = 0; input[i] != '\0'; i++) {
        checkOperator(input[i]);
    }

    return 0;
}
```

OUTPUT:


Samikshadhanya/CSA1447-COMP
Online C Compiler - Programiz
programiz.com/c-programming/online-compiler/
Gmail
YouTube
All Bookmarks
Programiz
C Online Compiler
Premium Coding Courses by Programiz
Programiz PRO
main.c
Run
Output
Clear
Search
ENG US
13:50:15
13-02-2025

```
    }
}

int main() {
    char input[100];
    printf("Enter an expression: ");
    fgets(input, sizeof(input), stdin);

    for (int i = 0; input[i] != '\0'; i++) {
        checkOperator(input[i]);
    }

    return 0;
}
```

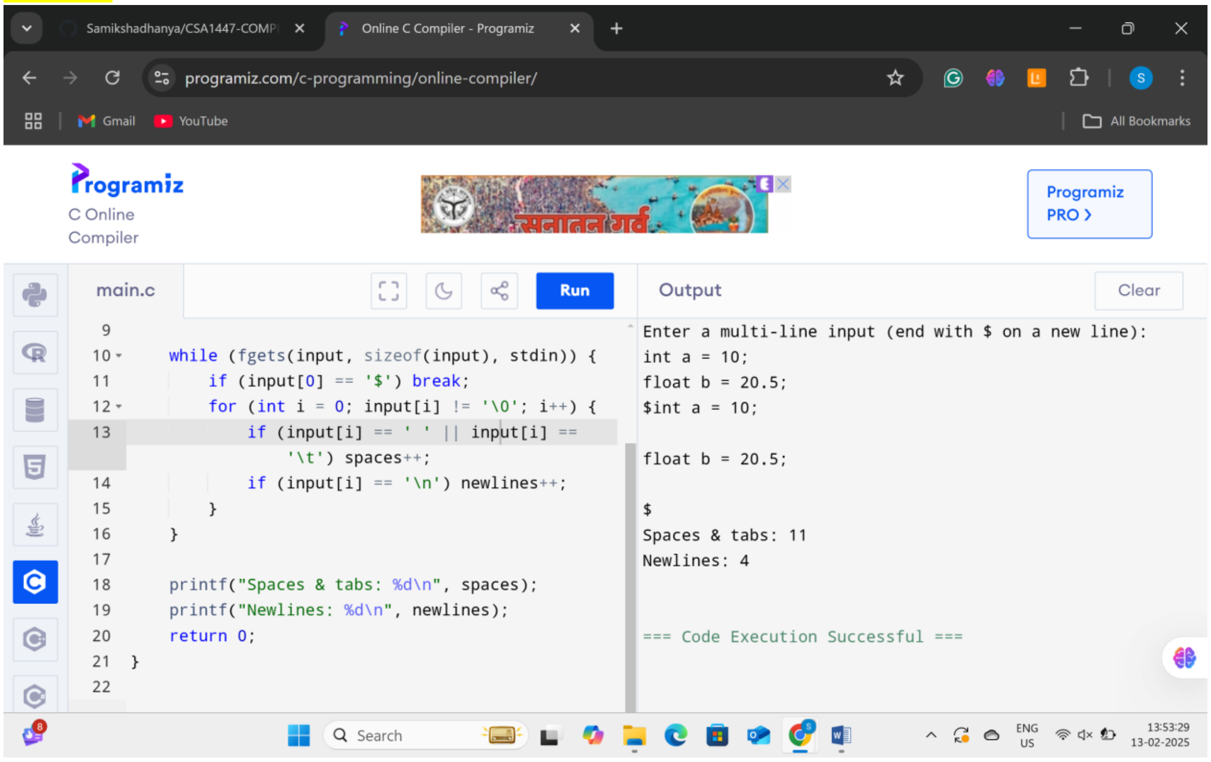
```
Enter an expression: Enter an expression: a + b - c * d /
 e

Operator: +
Operator: -
Operator: *
Operator: /

=== Code Execution Successful ===
```

EXPERIMENT-4

PROGRAM:

```
#include <stdio.h>

#include <string.h>


int main() {

   char input[200];

   int spaces = 0, newlines = 0;


   printf("Enter a multi-line input (end with $ on a new line):\n");


   while (fgets(input, sizeof(input), stdin)) {

      if (input[0] == '$') break;

      for (int i = 0; input[i] != '\0'; i++) {

         if (input[i] == ' ' || input[i] == '\t') spaces++;

         if (input[i] == '\n') newlines++;}}

   printf("Spaces & tabs: %d\n", spaces);

   printf("Newlines: %d\n", newlines);

   return 0;}
```

OUTPUT:


Samikshadhanya/CSA1447-COMP
Online C Compiler - Programiz
programiz.com/c-programming/online-compiler/
Gmail
YouTube
All Bookmarks
Programiz
C Online Compiler
Programiz PRO
main.c
Run
Output
Clear

```c

    while (fgets(input, sizeof(input), stdin)) {
        if (input[0] == '$') break;
        for (int i = 0; input[i] != '\0'; i++) {
            if (input[i] == ' ' || input[i] == '\t') spaces++;
            if (input[i] == '\n') newlines++;
        }
    }

    printf("Spaces & tabs: %d\n", spaces);
    printf("Newlines: %d\n", newlines);
    return 0;
}

```

```
Enter a multi-line input (end with $ on a new line):
int a = 10;
float b = 20.5;
$int a = 10;

float b = 20.5;


$
Spaces & tabs: 11
Newlines: 4


=== Code Execution Successful ===
```

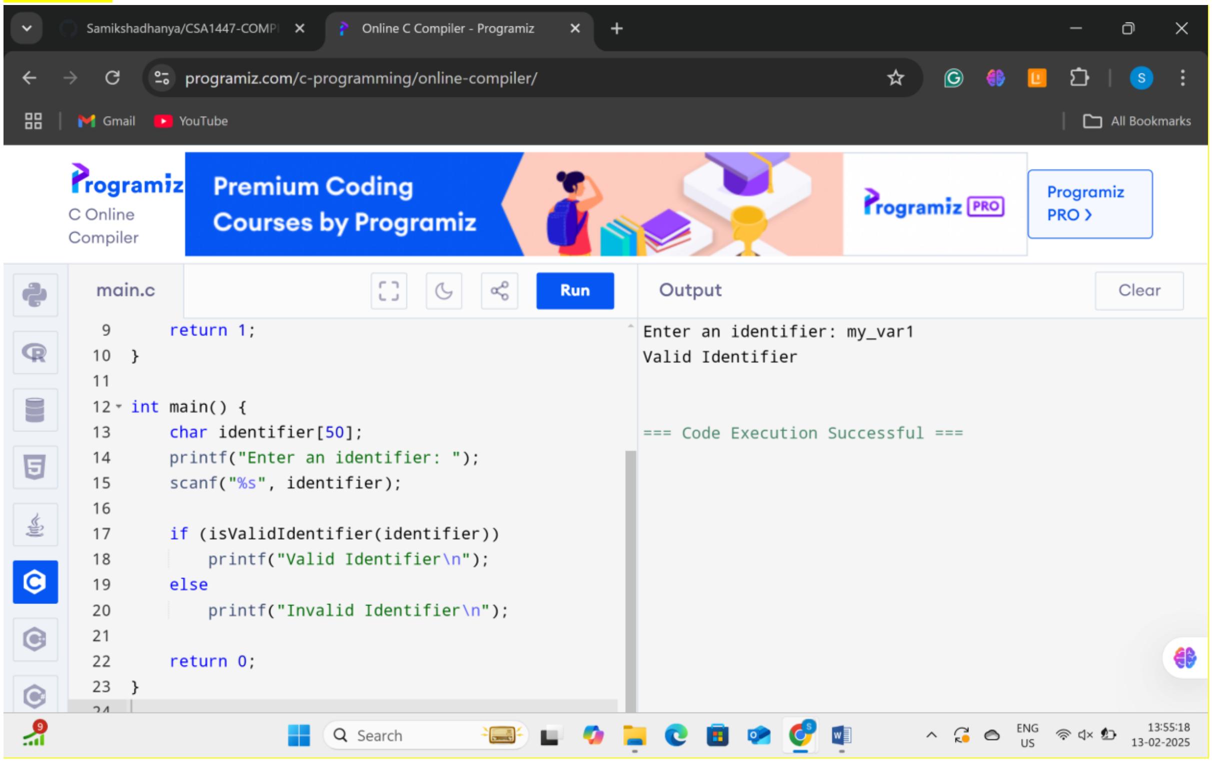
EXPERIMENT-5

PROGRAM:

```c
#include <stdio.h>
#include <ctype.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] != '_') return 0;
    }
    return 1;
}

int main() {
    char identifier[50];
    printf("Enter an identifier: ");
```

```
  scanf("%s", identifier);

  if (isValidIdentifier(identifier))
     printf("Valid Identifier\n");
  else
     printf("Invalid Identifier\n");
  return 0;
}
```

OUTPUT:


Samikshadhanya/CSA1447-COMP
Online C Compiler - Programiz
programiz.com/c-programming/online-compiler/
Gmail
YouTube
All Bookmarks
Programiz
C Online Compiler
Premium Coding Courses by Programiz
Programiz PRO
Programiz PRO ›
main.c
Run
Output
Clear
Search
ENG US
13:55:18
13-02-2025

```
        return 1;
    }

    int main() {
        char identifier[50];
        printf("Enter an identifier: ");
        scanf("%s", identifier);

        if (isValidIdentifier(identifier))
            printf("Valid Identifier\n");
        else
            printf("Invalid Identifier\n");

        return 0;
    }
```

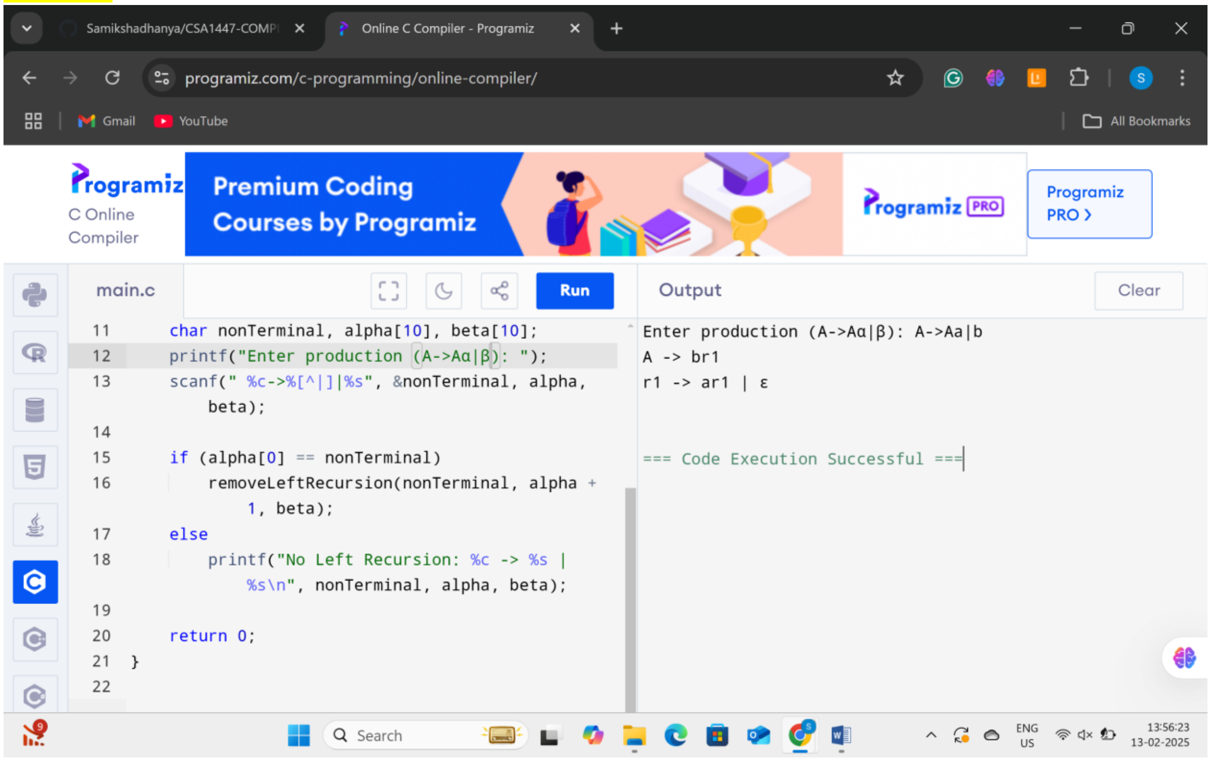
```
Enter an identifier: my_var1
Valid Identifier


=== Code Execution Successful ===
```

EXPERIMENT-6

PROGRAM:

```
#include <stdio.h>
#include <string.h>
void removeLeftRecursion(char nonTerminal, char alpha[], char beta[]) {
  char newNonTerminal = nonTerminal + '1';
  printf("%c -> %s%c1\n", nonTerminal, beta, newNonTerminal);
  printf("%c1 -> %s%c1 | ε\n", newNonTerminal, alpha, newNonTerminal);
}int main() {
  char nonTerminal, alpha[10], beta[10];

  printf("Enter production (A->Aα|β): ");

  scanf(" %c->%[^|]|%s", &nonTerminal, alpha, beta);

  if (alpha[0] == nonTerminal)

    removeLeftRecursion(nonTerminal, alpha + 1, beta);

  else

    printf("No Left Recursion: %c -> %s | %s\n", nonTerminal, alpha, beta);

  return 0;}
```

OUTPUT:


Samikshadhanya/CSA1447-COMP
Online C Compiler - Programiz
programiz.com/c-programming/online-compiler/
Gmail
YouTube
All Bookmarks
Programiz
C Online Compiler
Premium Coding Courses by Programiz
Programiz PRO
Programiz PRO ›
main.c
Run
Output
Clear
Search
ENG US
13:56:23
13-02-2025

```
    char nonTerminal, alpha[10], beta[10];
    printf("Enter production (A->Aα|β): ");
    scanf(" %c->%[^|]|%s", &nonTerminal, alpha,
        beta);

    if (alpha[0] == nonTerminal)
        removeLeftRecursion(nonTerminal, alpha +
            1, beta);
    else
        printf("No Left Recursion: %c -> %s |
            %s\n", nonTerminal, alpha, beta);

    return 0;
}
```

```
Enter production (A->Aα|β): A->Aa|b
A -> br1
r1 -> ar1 | ε


=== Code Execution Successful ===
```

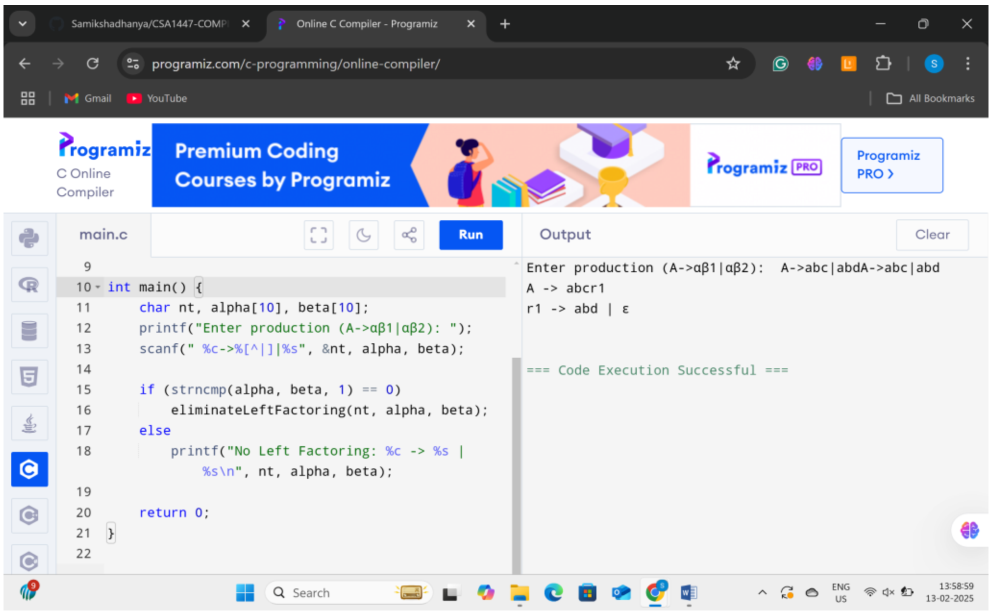
EXPERIMENT-7

PROGRAM:

```
#include <stdio.h>

#include <string.h>

void eliminateLeftFactoring(char nt, char alpha[], char beta[]) {

  char newNT = nt + '1';

  printf("%c -> %s%c1\n", nt, alpha, newNT);

  printf("%c1 -> %s | ε\n", newNT, beta);

}int main() {
```

```
char nt, alpha[10], beta[10];
printf("Enter production (A->αβ1|αβ2): ");
scanf(" %c->%[^|]|%s", &nt, alpha, beta);
if (strncmp(alpha, beta, 1) == 0)
   eliminateLeftFactoring(nt, alpha, beta);
else
   printf("No Left Factoring: %c -> %s | %s\n", nt, alpha, beta);
return 0;}
```

OUTPUT:


Samikshadhanya/CSA1447-COMP
Online C Compiler - Programiz
programiz.com/c-programming/online-compiler/
Gmail
YouTube
All Bookmarks
Programiz
C Online Compiler
Premium Coding Courses by Programiz
Programiz PRO
main.c
Run
Output
Clear
Search
ENG US
13:58:59
13-02-2025

```
int main() {
    char nt, alpha[10], beta[10];
    printf("Enter production (A->αβ1|αβ2): ");
    scanf(" %c->%[^|]|%s", &nt, alpha, beta);

    if (strncmp(alpha, beta, 1) == 0)
        eliminateLeftFactoring(nt, alpha, beta);
    else
        printf("No Left Factoring: %c -> %s |
            %s\n", nt, alpha, beta);

    return 0;
}
```

```
Enter production (A->αβ1|αβ2):  A->abc|abdA->abc|abd
A -> abcr1
r1 -> abd | ε


=== Code Execution Successful ===
```