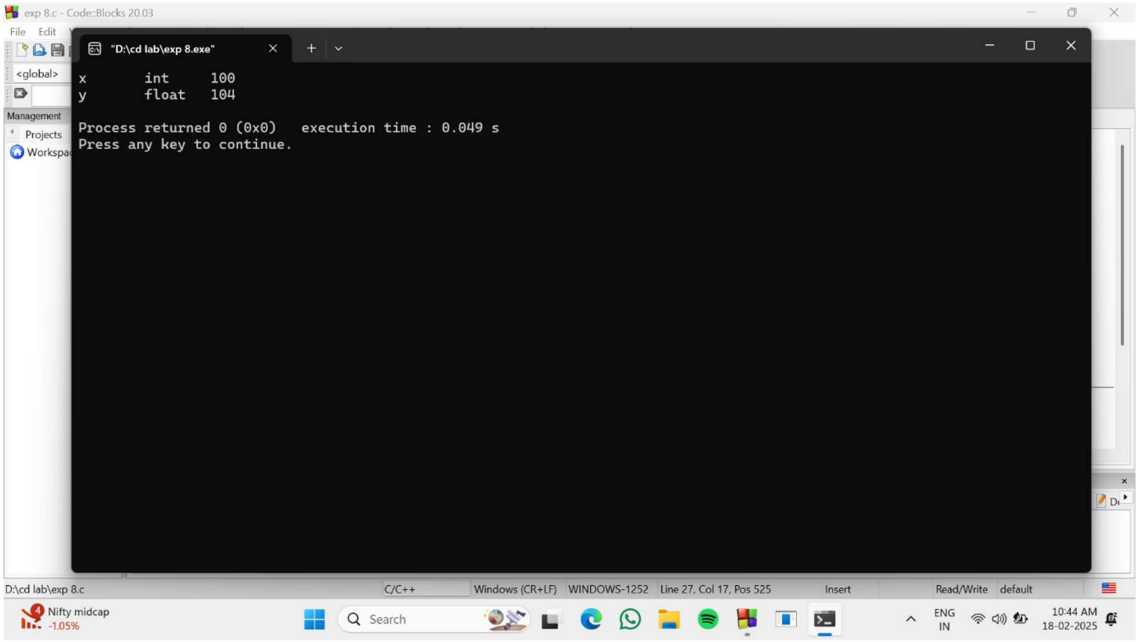
Experiment – 8

```
#include <stdio.h>
#include <string.h>
struct Symbol {
   char name[20], type[10];
   int address;
};
struct Symbol table[100];
int count = 0;
void insert(char name[], char type[], int address) {
   strcpy(table[count].name, name);
   strcpy(table[count].type, type);
   table[count].address = address;
   count++;
}
int search(char name[]) {
   for (int i = 0; i < count; i++) {
      if (strcmp(table[i].name, name) == 0)
         return i;
   }
   return -1;
}
void display() {
   for (int i = 0; i < count; i++) {
      printf("%s\t%s\t%d\n", table[i].name, table[i].type, table[i].address);
   }
}
int main() {
   insert("x", "int", 100);
```

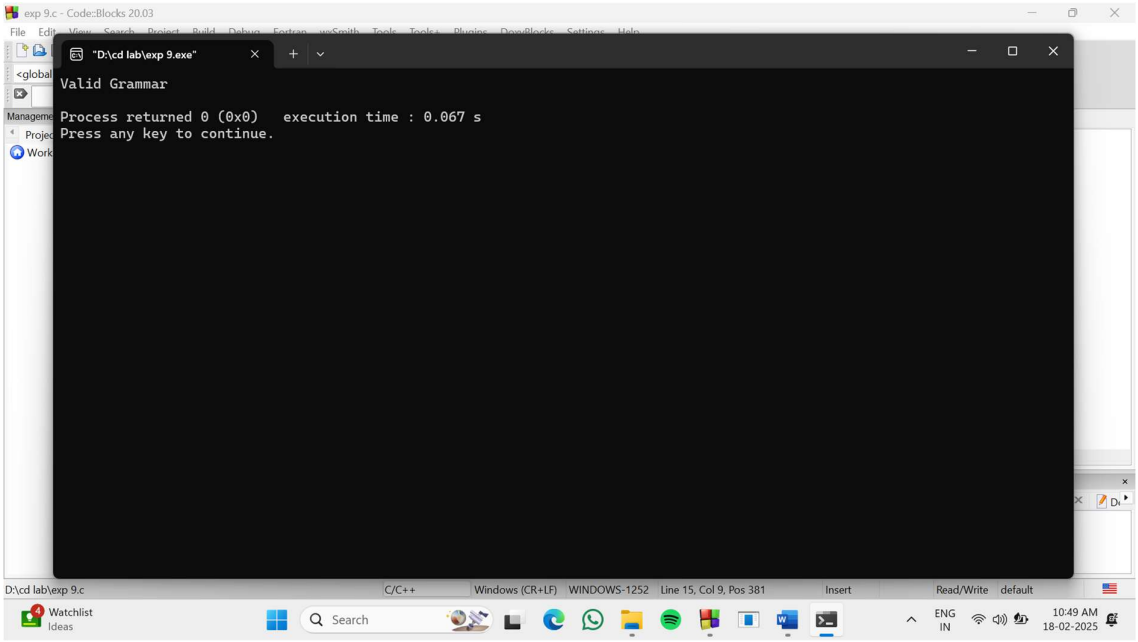
```
    insert("y", "float", 104);
    display();
    return 0;
}
```



Experiment – 9

```
#include <stdio.h>
#include <string.h>
int checkGrammar(char str[], int start, int end) {
    if (start > end) return 1;
    if (str[start] == 'a' && str[end] == 'b')
        return checkGrammar(str, start + 1, end - 1);
    return 0;
}
int main() {
    char str[] = "aabb";
    if (checkGrammar(str, 0, strlen(str) - 1))
        printf("Valid Grammar\n");
```

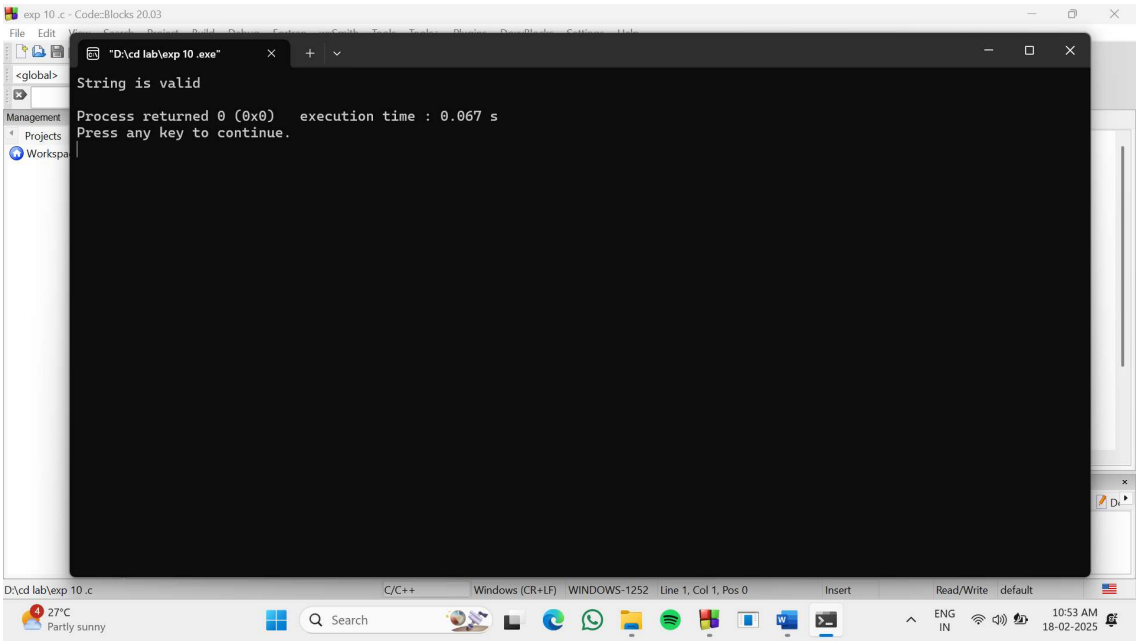
```
    else
        printf("Invalid Grammar\n");
    return 0;
}
```



Experiment – 10

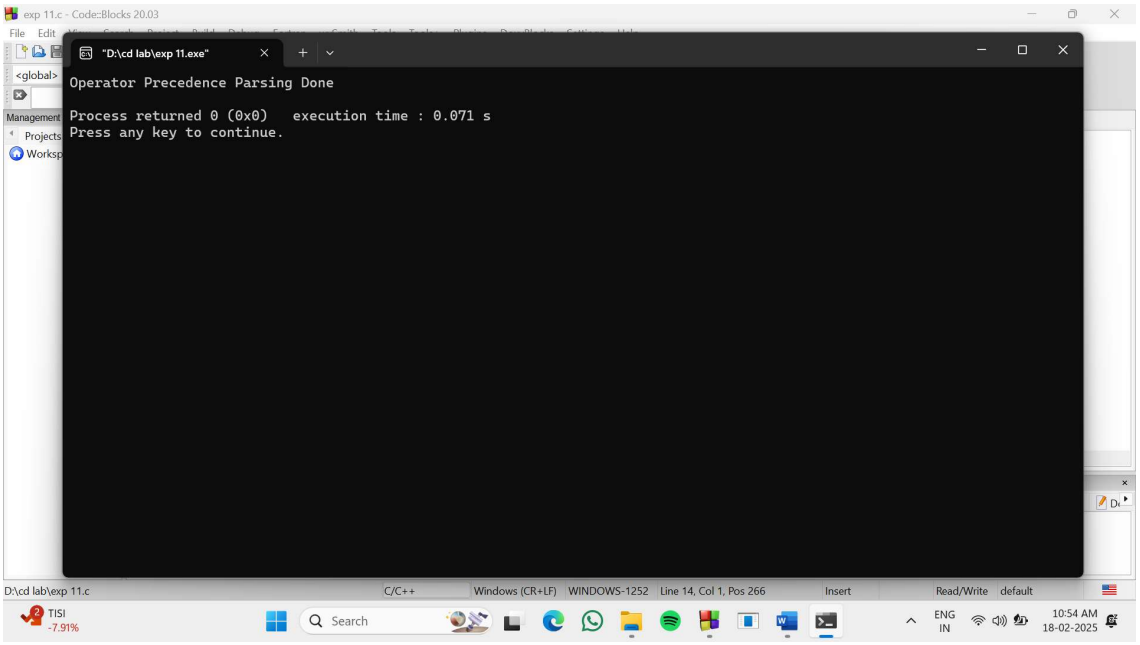
```
#include <stdio.h>
#include <string.h>
char *input;
void E(), T();
void E() {
    if (*input == 'a') {
        input++;
        T();
    }
}
void T() {
```

```
    if (*input == 'b') {
        input++;
    }
}
int main() {
    char str[] = "ab";
    input = str;
    E();
    if (*input == '\0')
        printf("String is valid\n");
    else
        printf("String is invalid\n");
    return 0;
}
```



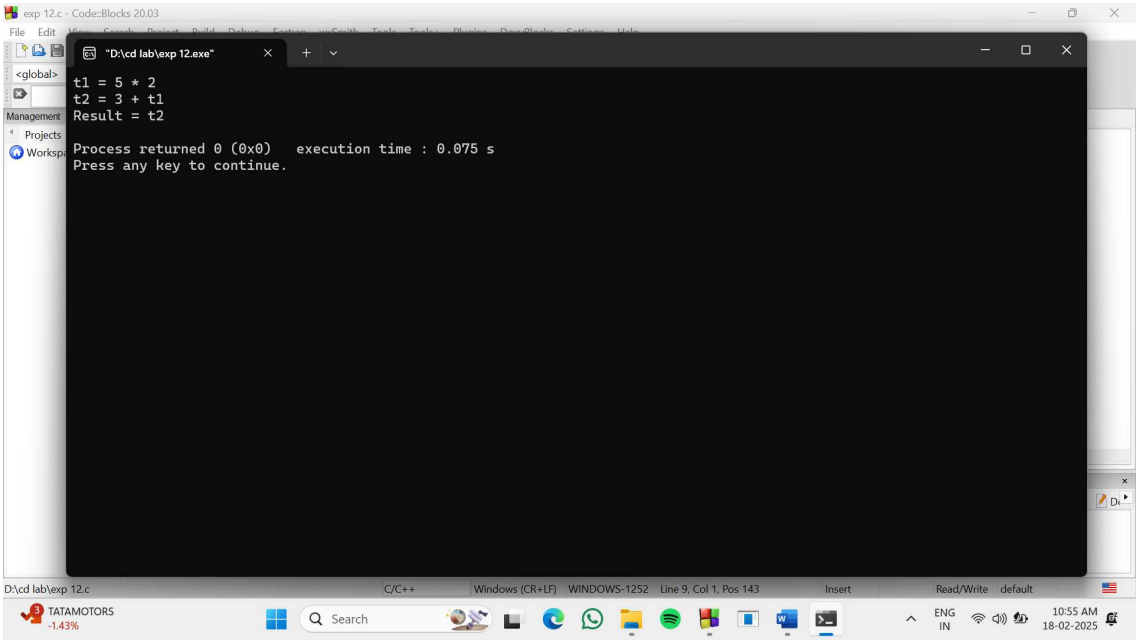
Experiment – 11

```
#include <stdio.h>
int precedence(char op) {
    if (op == '+' || op == '-') return 1;
    if (op == '*' || op == '/') return 2;
    return 0;
}
int main() {
    char expr[] = "3+5*2";
    printf("Operator Precedence Parsing Done\n");
    return 0;
}
```



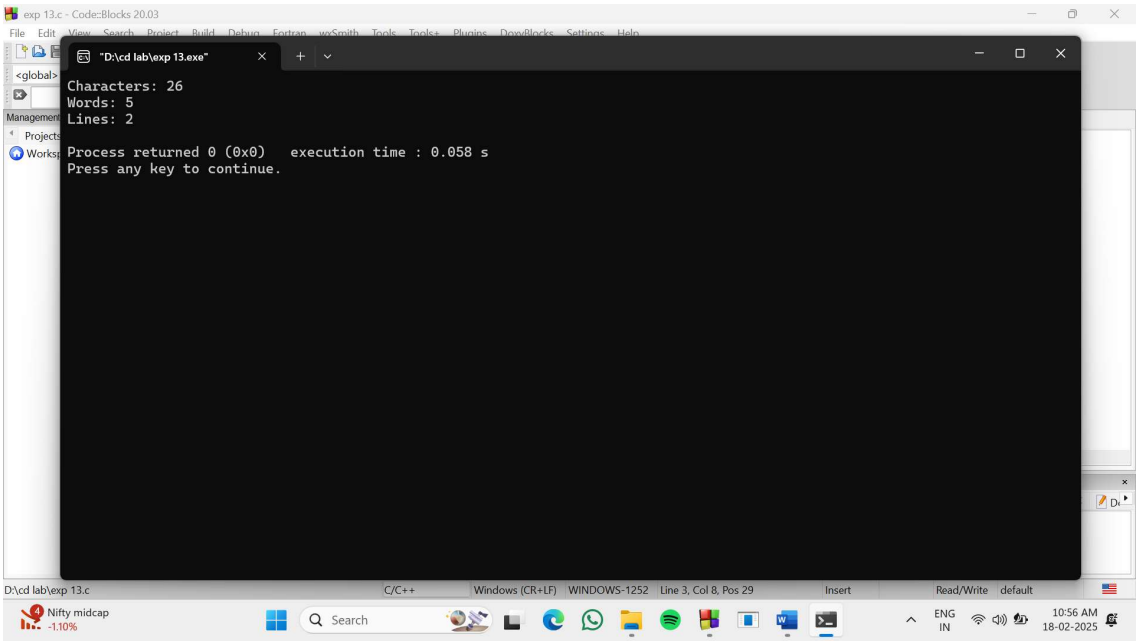
Experiment – 12

```
#include <stdio.h>

int main() {
    printf("t1 = 5 * 2\n");
    printf("t2 = 3 + t1\n");
    printf("Result = t2\n");
    return 0;
}
```



Experiment – 13

```
#include <stdio.h>

int main() {
    char str[] = "Hello World\nThis is a test";
    int chars = 0, words = 1, lines = 1;
    for (int i = 0; str[i] != '\0'; i++) {
        chars++;
        if (str[i] == ' ') words++;
        if (str[i] == '\n') lines++;
    }
    printf("Characters: %d\nWords: %d\nLines: %d\n", chars, words, lines);
    return 0;
}
```



Experiment – 14

```
#include <stdio.h>

int main() {
    int a = 5, b = 2;
    int t1 = a * b;
    int x = t1 + 3;
    int y = t1 + 4;
    printf("x = %d, y = %d\n", x, y);
    return 0;
}
```