Online C Compiler - Progra | Online C Compiler | Online C Compiler | Online C Compiler | Online C Compiler | Online C Compiler | Online C Compiler | Online C Compiler

www.programiz.com/c-programming/online-compiler/

Data Analytics usi... | Python Programs |... | Futurepedia - Find... | Career Essentials i... | Collections | Micr... | TensorFlow - Core... | Your Work - Code... | Login | Skills for Al... | Introduction to Cy...

Programiz

C Online Compiler

Programiz PRO >

main.c

Share | Run

```c
// question 1
#include <stdio.h>
#include <string.h>
#include <ctype.h>

// Define token types
#define ID 1=
#define CONSTANT 2
#define OPERATOR 3
#define COMMENT 4

// Maximum length of an identifier
#define MAX_ID_LENGTH 32

// Function to check if a character is a letter or underscore
int is_letter_or_underscore(char c) {
    return (isalpha(c) || c == '_');
}

// Function to check if a character is a digit
int is_digit(char c) {
    return (isdigit(c));
}

// Function to check if a string is an identifier
```

Output | Clear

```
Enter input: x = 5
x = 5

Identifier: x
Operator: =
Constant: 5


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

88°F Mostly sunny | Search | ENG IN | 2:34 PM 2/10/2025

Online C Compiler | Online C Compiler - Progra | Online C Compiler | Online C Compiler | Online C Compiler | Online C Compiler | Online C Compiler | Online C Compiler

www.programiz.com/c-programming/online-compiler/

Data Analytics usi... | Python Programs |... | Futurepedia - Find... | Career Essentials i... | Collections | Micr... | TensorFlow - Core... | Your Work - Code... | Login | Skills for Al... | Introduction to Cy...

Programiz

C Online Compiler

Programiz PRO >

main.c

Share | Run

```c
//question 2
#include <stdio.h>
#include <string.h>
#include <ctype.h>

// Define token types
#define ID 1
#define CONSTANT 2
#define OPERATOR 3
#define COMMENT 4

// Maximum length of an identifier
#define MAX_ID_LENGTH 32

// Function to check if a character is a letter or underscore
int is_letter_or_underscore(char c) {
    return (isalpha(c) || c == '_');
}

// Function to check if a character is a digit
int is_digit(char c) {
    return (isdigit(c));
}

// Function to check if a string is an identifier
```

Output

Clear

```
Enter input: x = 5 + 3 * y
x = 5 + 3 * y

Identifier: x
Operator: =
Constant: 5
Operator: +
Constant: 3
Operator: *
Identifier: y


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

88°F Mostly sunny | Search | ENG IN | 2:34 PM 2/10/2025

Online C Compiler | Online C Compiler | Online C Compiler - Progra | Online C Compiler | Online C Compiler | Online C Compiler | Online C Compiler | Online C Compiler

www.programiz.com/c-programming/online-compiler/

Data Analytics usi... | Python Programs |... | Futurepedia - Find... | Career Essentials i... | Collections | Micr... | TensorFlow - Core... | Your Work - Code... | Login | Skills for Al... | Introduction to Cy...

Programiz
C Online Compiler

Programiz PRO ›

main.c

Share | Run

```c
//question 3
#include <stdio.h>
#include <string.h>

int is_operator(char c) {
    char operators[] = {'+', '-', '*', '/'};
    for (int i = 0; i < 4; i++) {
        if (c == operators[i]) {
            return 1;
        }
    }
    return 0;
}

int main() {
    char input[1024];
    printf("Enter input: ");
    fgets(input, 1024, stdin);

    for (int i = 0; i < strlen(input); i++) {
        if (is_operator(input[i])) {
            printf("Operator '%c' recognized.\n", input[i]);
        }
    }

```

Output | Clear

```
Enter input: x*x*=8
Operator '*' recognized.
Operator '*' recognized.


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

88°F
Mostly sunny

Search

ENG
IN

2:34 PM
2/10/2025

Online C Compiler - Online C Compiler - Online C Compiler - Online C Compiler - Progra Online C Compiler - Online C Compiler - Online C Compiler - Online C Compiler -

www.programiz.com/c-programming/online-compiler/

Data Analytics usi... Python Programs |... Futurepedia - Find... Career Essentials i... Collections | Micr... TensorFlow - Core... Your Work - Code... Login | Skills for Al... Introduction to Cy...

Programiz

C Online Compiler

Programiz PRO ›

main.c

Share Run

Output Clear

```c
//question 4
#include <stdio.h>
#include <string.h>
#include <ctype.h>

int main() {
    char input[1024];
    int whitespace_count = 0;
    int newline_count = 0;

    printf("Enter input: ");
    fgets(input, 1024, stdin);

    for (int i = 0; i < strlen(input); i++) {
        if (isspace(input[i])) {
            whitespace_count++;
            if (input[i] == '\n') {
                newline_count++;
            }
        }
    }

    printf("Number of whitespaces: %d\n", whitespace_count);
    printf("Number of newline characters: %d\n", newline_count);

```

```
Enter input: Hello World!
Hello World!

Number of whitespaces: 2
Number of newline characters: 1


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

88°F Mostly sunny

Search

ENG IN

2:37 PM 2/10/2025

Online C Compiler - | Online C Compiler - | Online C Compiler - | Online C Compiler - | Online C Compiler - Progra | Online C Compiler - | Online C Compiler - | Online C Compiler -

www.programiz.com/c-programming/online-compiler/

Data Analytics usi... | Python Programs |... | Futurepedia - Find... | Career Essentials i... | Collections | Micr... | TensorFlow - Core... | Your Work - Code... | Login | Skills for Al... | Introduction to Cy...

Programiz
C Online Compiler

Programiz PRO >

main.c

Share | Run

```c
//question 5
#include <stdio.h>
#include <ctype.h>
#include <string.h>

// Function to check if an identifier is valid
int is_valid_identifier(char* identifier) {
    // Check if the identifier is empty
    if (strlen(identifier) == 0) {
        return 0;
    }

    // Check if the first character is a letter or underscore
    if (!isalpha(identifier[0]) && identifier[0] != '_') {
        return 0;
    }

    // Check if the remaining characters are letters, digits, or underscores
    for (int i = 1; i < strlen(identifier); i++) {
        if (!isalpha(identifier[i]) && !isdigit(identifier[i]) &&
            identifier[i] != '_') {
            return 0;
        }
    }

```

Output | Clear

```
Enter an identifier: hello_world
hello_world is a valid identifier.


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

88°F Mostly sunny | Search | ENG IN | 2:38 PM 2/10/2025

Online C Compiler - Progra

www.programiz.com/c-programming/online-compiler/

Programiz

C Online Compiler

Programiz PRO ›

main.c

Share

Run

Output

Clear

```c
// question 6
#include <stdio.h>
#include <string.h>
#include <stdlib.h>

// Define the structure for a production rule
typedef struct {
    char non_terminal;
    char* production;
} Production;

// Function to eliminate left recursion
void eliminate_left_recursion(Production* productions, int num_productions)
    {
    for (int i = 0; i < num_productions; i++) {
        char non_terminal = productions[i].non_terminal;
        char* production = productions[i].production;

        // Check if the production is left recursive
        if (production[0] == non_terminal) {
            // Eliminate left recursion
            printf("Left recursion detected for non-terminal '%c'\n",
                non_terminal);

            // Find all non-left-recursive productions for the same non
```

```
Left recursion detected for non-terminal 'E'
New production rules:
E -> TE'
E' -> +TE' | epsilon
Left recursion detected for non-terminal 'T'
New production rules:
T -> FT'
T' -> *FT' | epsilon


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

88°F Mostly sunny

Search

ENG IN

2:38 PM 2/10/2025

Online C Compiler - Online C Compiler - Online C Compiler - Online C Compiler - Online C Compiler - Online C Compiler - Online C Compiler - Online C Compiler -

www.programiz.com/c-programming/online-compiler/

Data Analytics usi... | Python Programs |... | Futurepedia - Find... | Career Essentials i... | Collections | Micr... | TensorFlow - Core... | Your Work - Code... | Login | Skills for Al... | Introduction to Cy...

Programiz
C Online Compiler

Programiz PRO >

main.c | Share | Run

```c
// question 7
#include <stdio.h>
#include <string.h>
#include <stdlib.h>

// Define the structure for a production rule
typedef struct {
    char non_terminal;
    char* production;
} Production;

// Function to eliminate left factoring
void eliminate_left_factoring(Production* productions, int num_productions)
    {
    for (int i = 0; i < num_productions; i++) {
        char non_terminal = productions[i].non_terminal;
        char* production = productions[i].production;

        // Find all productions with the same non-terminal and common prefix
        char common_prefix[100];
        int common_prefix_length = 0;
        int num_common_productions = 0;

        for (int j = 0; j < num_productions; j++) {
            if (productions[j].non_terminal == non_terminal && strcmp
```

Output | Clear

```
Left factoring detected for non-terminal 'E'
New production rules:
E -> TE'
E' -> +F | epsilon
E' -> *F | epsilon
Left factoring detected for non-terminal 'E'
New production rules:
E -> TE'
E' -> +F | epsilon
E' -> *F | epsilon
Left factoring detected for non-terminal 'T'
New production rules:
T -> TT'
T' -> T*F | epsilon
T' -> F | epsilon
Left factoring detected for non-terminal 'T'
New production rules:
T -> TT'
T' -> T*F | epsilon
T' -> F | epsilon
Left factoring detected for non-terminal 'F'
New production rules:
F -> TF'
F' -> (E) | epsilon
F' -> id | epsilon
```

88°F Mostly sunny | Search | ENG IN | 2:39 PM 2/10/2025

Online C Compiler - | Online C Compiler - | Online C Compiler - | Online C Compiler - | Online C Compiler - | Online C Compiler - | Online C Compiler - | Online C Compiler - Progra

www.programiz.com/c-programming/online-compiler/

Data Analytics usi... | Python Programs |... | Futurepedia - Find... | Career Essentials i... | Collections | Micr... | TensorFlow - Core... | Your Work - Code... | Login | Skills for Al... | Introduction to Cy...

Programiz

C Online Compiler

Programiz PRO >

main.c

Share | Run

```c
// question 8
#include <stdio.h>
#include <string.h>
#include <stdlib.h>

// Define the structure for a symbol table entry
typedef struct {
    char name[100];
    char type[100];
    int size;
} Symbol;

// Define the structure for a symbol table
typedef struct {
    Symbol* entries;
    int capacity;
    int size;
} SymbolTable;

// Function to create a new symbol table
SymbolTable* create_symbol_table(int capacity) {
    SymbolTable* table = (SymbolTable*)malloc(sizeof(SymbolTable));
    table->entries = (Symbol*)malloc(capacity * sizeof(Symbol));
    table->capacity = capacity;
    table->size = 0;
```

Output | Clear

```
Symbol 'x' inserted successfully.
Symbol 'y' inserted successfully.
Symbol 'z' inserted successfully.
Symbol Table:
x (int, 4)
y (float, 8)
z (char, 1)
Symbol 'y' found in the symbol table.
Type: float, Size: 8
Symbol 'x' deleted successfully.
Symbol Table:
y (float, 8)
z (char, 1)


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

88°F Mostly sunny | Search | ENG IN | 2:39 PM 2/10/2025