

1. Variable Initialization

Question: Write a program that declares an integer variable, initializes it with a value of 42, and prints the value to the console.

The screenshot shows a web browser window with the URL `onlinegdb.com/?`. The browser's address bar and tabs are visible at the top. The main content area displays a C program in a dark-themed editor. The program is titled `main.c` and contains the following code:

```
1- /******  
2  
3 1. Variable Initialization  
4 Question: Write a program that declares an integer variable,  
5 initializes it with a value of 42, and prints the value to the console.  
6  
7 *****/  
8 #include <stdio.h>  
9  
10 int main()  
11 {  
12     int a; //signed integer variable  
13     a=42;  
14     printf("The value of the variable a is = %d \n",a);  
15  
16  
17     return 0;  
18 }  
19  
20
```

Below the code editor, there is a console window showing the output of the program:

```
input  
The value of the variable a is = 42  
  
...Program finished with exit code 0  
Press ENTER to exit console.
```

On the right side of the editor, there is a sidebar with a "Free ChatGPT Extension" advertisement. The sidebar also contains a "close ad [x]" button. The bottom of the browser window shows the Windows taskbar with various application icons and the system clock displaying 11:07 on 08-11-2024.

2. Swapping Variables

Question: Create a program that swaps the values of two integer variables without using a temporary variable. Demonstrate this by printing the values before and after the swap.

The screenshot shows a web browser window with the URL `onlinegdb.com/?`. The browser's address bar and tabs are visible at the top. The main content area displays a C program in a dark-themed editor. The program is titled `main.c` and contains the following code:

```
2  
3 2. Swapping Variables  
4 Question: Create a program that swaps the values of two integer variables without using  
5 a temporary variable. Demonstrate this by printing the values before and after the swap.  
6  
7 *****/  
8 #include <stdio.h>  
9  
10 int main()  
11 {  
12     int a,b; //signed integer variable  
13     a=42;  
14     b=10;  
15     printf("The value of the variable a & b initially : a = %d , b = %d \n",a,b);  
16     //Swapping of a & b without temp var.  
17     a=a+b;  
18     b=a-b;  
19     a=a-b;  
20     printf("The value of the variable a & b after swapping : a = %d , b = %d \n",a,b);  
21  
22     return 0;  
23 }  
24
```

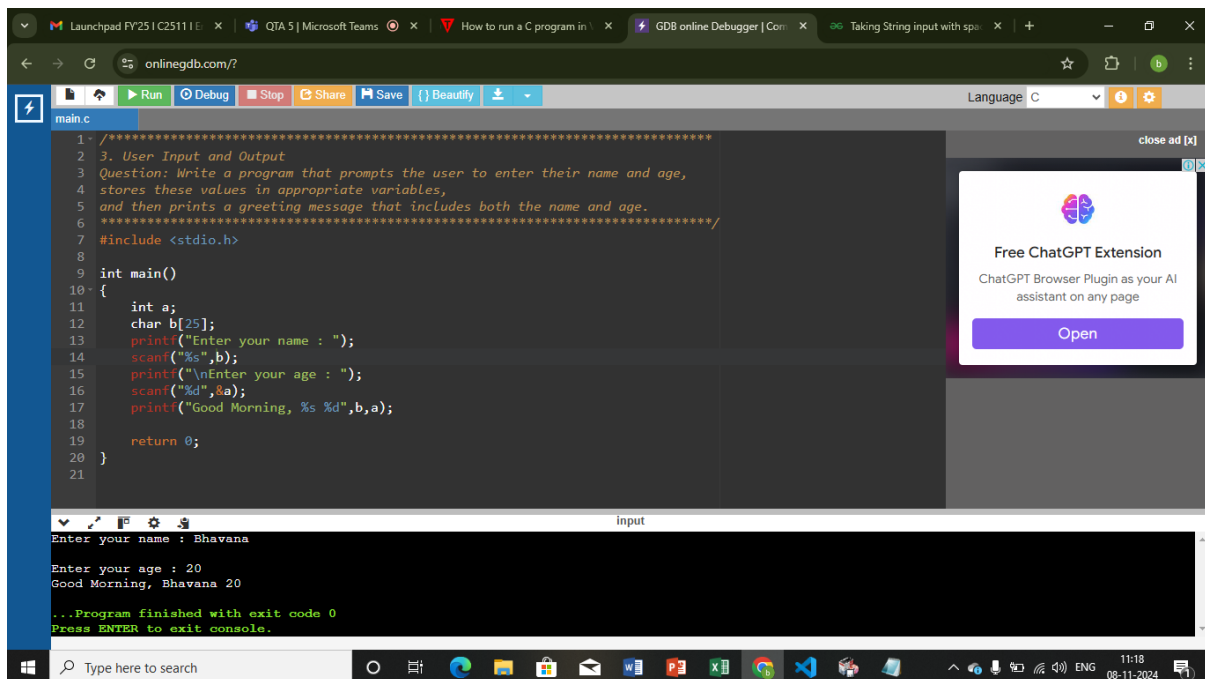
Below the code editor, there is a console window showing the output of the program:

```
input  
The value of the variable a & b initially : a = 42 , b = 10  
The value of the variable a & b after swapping : a = 10 , b = 42  
  
...Program finished with exit code 0  
Press ENTER to exit console.
```

On the right side of the editor, there is a sidebar with a "Free ChatGPT Extension" advertisement. The sidebar also contains a "close ad [x]" button. The bottom of the browser window shows the Windows taskbar with various application icons and the system clock displaying 11:10 on 08-11-2024.

3. User Input and Output

Question: Write a program that prompts the user to enter their name and age, stores these values in appropriate variables, and then prints a greeting message that includes both the name and age.



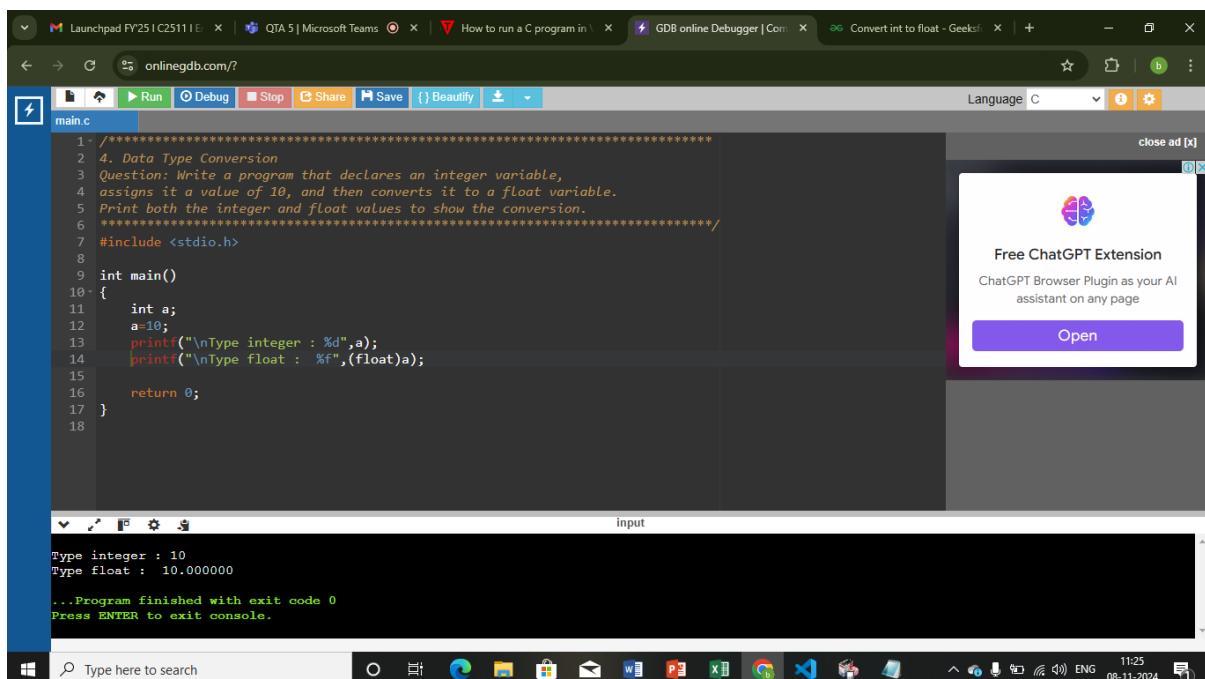
The screenshot shows a web browser window with the URL `onlinegdb.com/?`. The browser's address bar and tabs are visible at the top. The main content area displays a C program in a dark-themed editor. The program is a simple C program that prompts the user for their name and age, stores them in variables, and prints a greeting. The code is as follows:

```
1- /******  
2- 3. User Input and Output  
3- Question: Write a program that prompts the user to enter their name and age,  
4- stores these values in appropriate variables,  
5- and then prints a greeting message that includes both the name and age.  
6- *****/  
7- #include <stdio.h>  
8-  
9- int main()  
10- {  
11-     int a;  
12-     char b[25];  
13-     printf("Enter your name : ");  
14-     scanf("%s",b);  
15-     printf("\nEnter your age : ");  
16-     scanf("%d",&a);  
17-     printf("Good Morning, %s %d",b,a);  
18-  
19-     return 0;  
20- }  
21-
```

Below the code editor, there is an input field and a console output area. The input field shows the user has entered "Bhavana" for the name and "20" for the age. The console output shows the program's execution: "Enter your name : Bhavana", "Enter your age : 20", "Good Morning, Bhavana 20", and "...Program finished with exit code 0".

4. Data Type Conversion

Question: Write a program that declares an integer variable, assigns it a value of 10, and then converts it to a float variable. Print both the integer and float values to show the conversion.



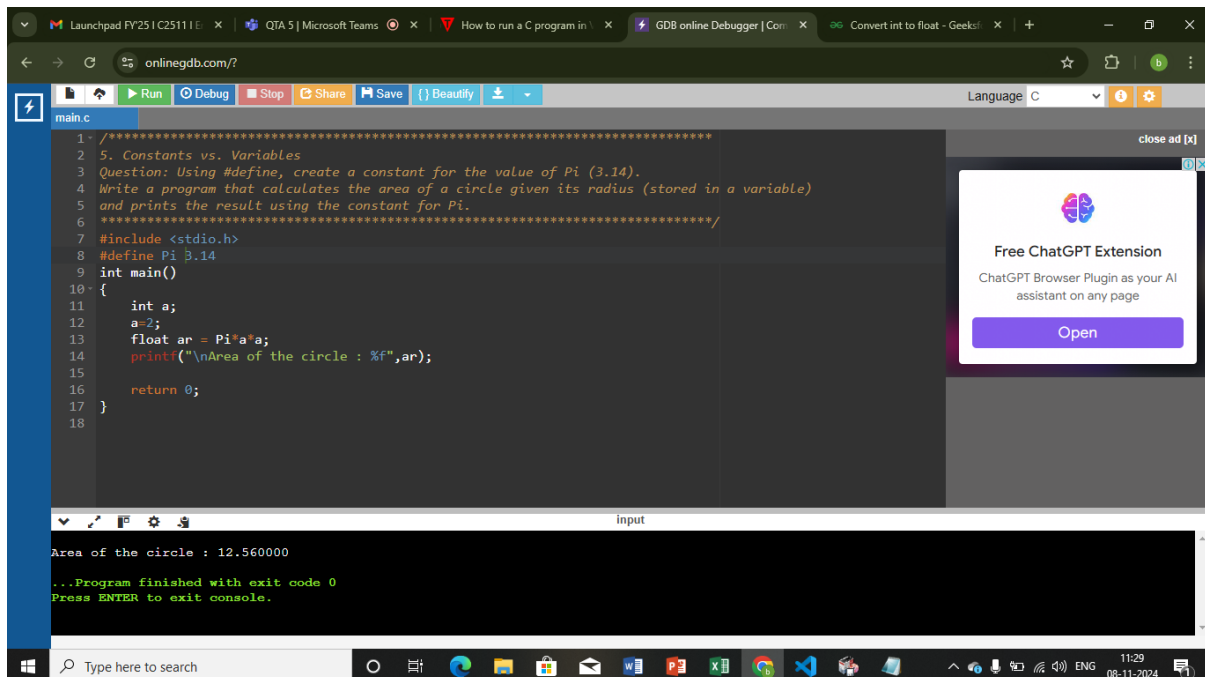
The screenshot shows a web browser window with the URL `onlinegdb.com/?`. The browser's address bar and tabs are visible at the top. The main content area displays a C program in a dark-themed editor. The program is a simple C program that declares an integer variable, assigns it a value of 10, and then converts it to a float variable. The code is as follows:

```
1- /******  
2- 4. Data Type Conversion  
3- Question: Write a program that declares an integer variable,  
4- assigns it a value of 10, and then converts it to a float variable.  
5- Print both the integer and float values to show the conversion.  
6- *****/  
7- #include <stdio.h>  
8-  
9- int main()  
10- {  
11-     int a;  
12-     a=10;  
13-     printf("\nType integer : %d",a);  
14-     printf("\nType float : %f",(float)a);  
15-  
16-     return 0;  
17- }  
18-
```

Below the code editor, there is an input field and a console output area. The input field is empty. The console output shows the program's execution: "Type integer : 10", "Type float : 10.000000", and "...Program finished with exit code 0".

5. Constants vs. Variables

Question: Using #define, create a constant for the value of Pi (3.14). Write a program that calculates the area of a circle given its radius (stored in a variable) and prints the result using the constant for Pi.



The screenshot shows a web browser window with the URL `onlinegdb.com/?`. The browser's address bar and tabs are visible at the top. The main content area displays a C program in a dark-themed editor. The program is titled `main.c` and contains the following code:

```
1- /******  
2 5. Constants vs. Variables  
3 Question: Using #define, create a constant for the value of Pi (3.14).  
4 Write a program that calculates the area of a circle given its radius (stored in a variable)  
5 and prints the result using the constant for Pi.  
6 *****/  
7 #include <stdio.h>  
8 #define Pi 3.14  
9 int main()  
10 {  
11     int a;  
12     a=2;  
13     float ar = Pi*a*a;  
14     printf("\nArea of the circle : %f",ar);  
15  
16     return 0;  
17 }  
18
```

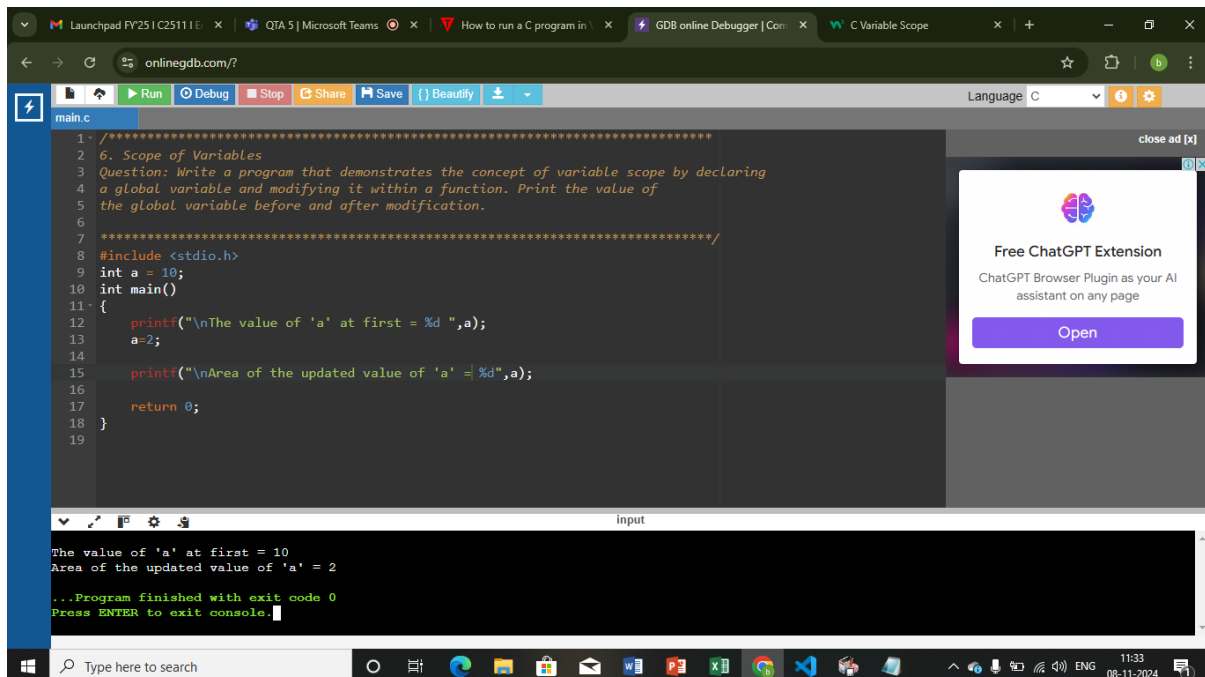
Below the code editor, there is an 'input' section and a console output area. The console output shows the result of the program execution:

```
Area of the circle : 12.560000  
...Program finished with exit code 0  
Press ENTER to exit console.
```

On the right side of the editor, there is a sidebar with a 'Free ChatGPT Extension' advertisement. The Windows taskbar is visible at the bottom of the screen.

6. Scope of Variables

Question: Write a program that demonstrates the concept of variable scope by declaring a global variable and modifying it within a function. Print the value of the global variable before and after modification.



The screenshot shows a web browser window with the URL `onlinegdb.com/?`. The browser's address bar and tabs are visible at the top. The main content area displays a C program in a dark-themed editor. The program is titled `main.c` and contains the following code:

```
1- /******  
2 6. Scope of Variables  
3 Question: Write a program that demonstrates the concept of variable scope by declaring  
4 a global variable and modifying it within a function. Print the value of  
5 the global variable before and after modification.  
6 *****/  
7 #include <stdio.h>  
8 int a = 10;  
9 int main()  
10 {  
11     printf("\nThe value of 'a' at first = %d ",a);  
12     a=2;  
13  
14     printf("\nArea of the updated value of 'a' = %d",a);  
15  
16     return 0;  
17 }  
18  
19
```

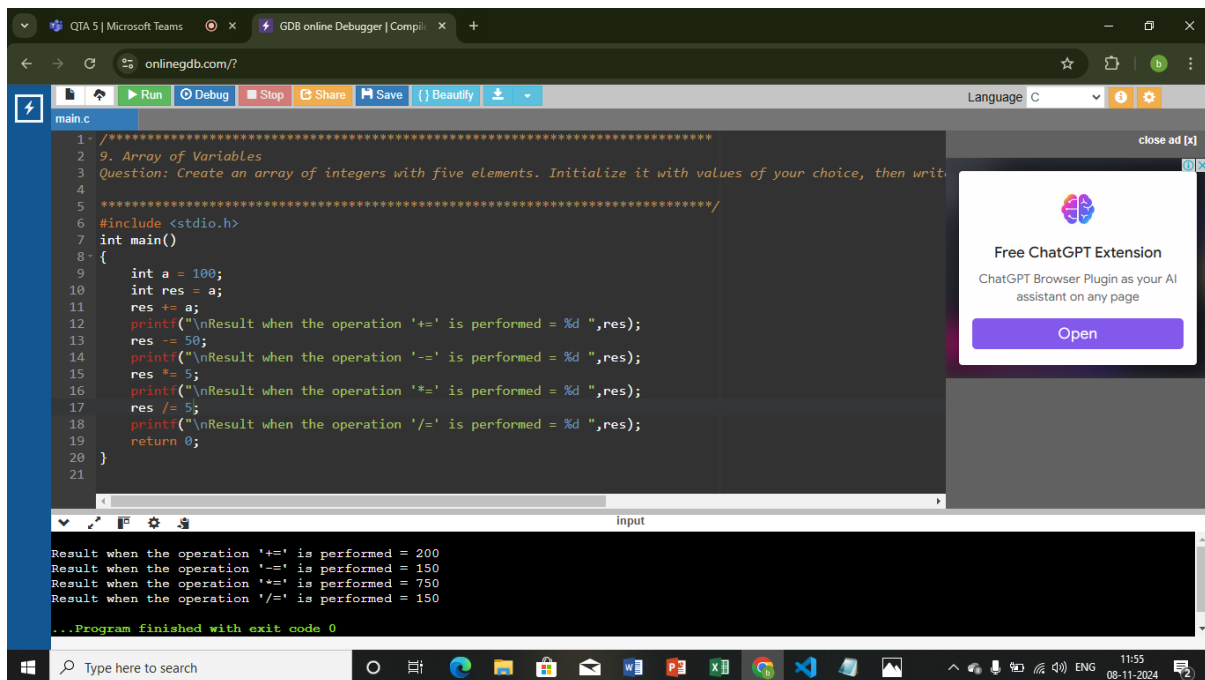
Below the code editor, there is an 'input' section and a console output area. The console output shows the result of the program execution:

```
The value of 'a' at first = 10  
Area of the updated value of 'a' = 2  
...Program finished with exit code 0  
Press ENTER to exit console.
```

On the right side of the editor, there is a sidebar with a 'Free ChatGPT Extension' advertisement. The Windows taskbar is visible at the bottom of the screen.

7. Using Augmented Assignment Operators

Question: Write a program that uses augmented assignment operators (+=, -=, *=, /=) to perform calculations on an integer variable initialized to 100. Print the value after each operation.



The screenshot shows a C program in a web-based IDE. The program initializes an integer variable 'a' to 100 and then performs four operations: addition, subtraction, multiplication, and division, each followed by a print statement. The output shows the results of these operations: 200, 150, 750, and 150. A 'Free ChatGPT Extension' pop-up is visible on the right side of the IDE.

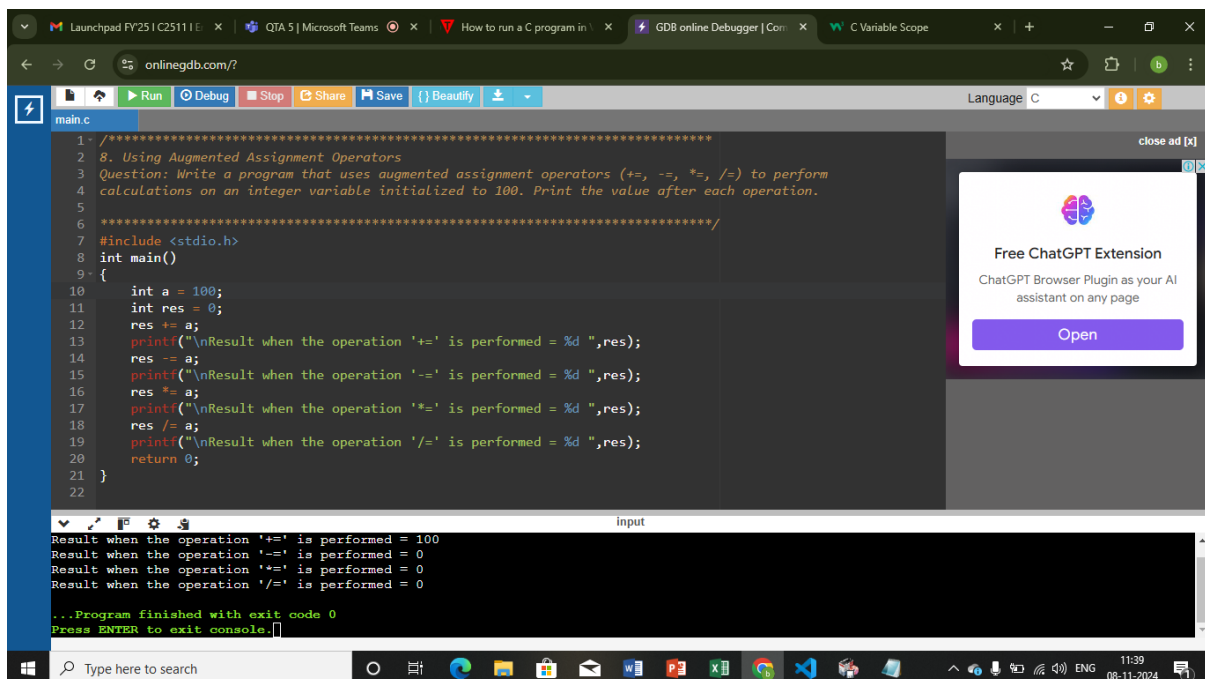
```
1- /******  
2 9. Array of Variables  
3 Question: Create an array of integers with five elements. Initialize it with values of your choice, then write  
4 *****/  
5  
6 #include <stdio.h>  
7 int main()  
8 {  
9     int a = 100;  
10    int res = a;  
11    res += a;  
12    printf("\nResult when the operation '+' is performed = %d ",res);  
13    res -= 50;  
14    printf("\nResult when the operation '-' is performed = %d ",res);  
15    res *= 5;  
16    printf("\nResult when the operation '*' is performed = %d ",res);  
17    res /= 5;  
18    printf("\nResult when the operation '/' is performed = %d ",res);  
19    return 0;  
20 }  
21
```

input

```
Result when the operation '+' is performed = 200  
Result when the operation '-' is performed = 150  
Result when the operation '*' is performed = 750  
Result when the operation '/' is performed = 150  
...Program finished with exit code 0
```

8. Array of Variables

Question: Create an array of integers with five elements. Initialize it with values of your choice, then write a program to calculate and print the sum of all elements in the array.



The screenshot shows a C program in a web-based IDE. The program initializes an integer variable 'a' to 100 and then performs four operations: addition, subtraction, multiplication, and division, each followed by a print statement. The output shows the results of these operations: 100, 0, 0, and 0. A 'Free ChatGPT Extension' pop-up is visible on the right side of the IDE.

```
1- /******  
2 8. Using Augmented Assignment Operators  
3 Question: Write a program that uses augmented assignment operators (+=, -=, *=, /=) to perform  
4 calculations on an integer variable initialized to 100. Print the value after each operation.  
5 *****/  
6  
7 #include <stdio.h>  
8 int main()  
9 {  
10    int a = 100;  
11    int res = 0;  
12    res += a;  
13    printf("\nResult when the operation '+' is performed = %d ",res);  
14    res -= a;  
15    printf("\nResult when the operation '-' is performed = %d ",res);  
16    res *= a;  
17    printf("\nResult when the operation '*' is performed = %d ",res);  
18    res /= a;  
19    printf("\nResult when the operation '/' is performed = %d ",res);  
20    return 0;  
21 }  
22
```

input

```
Result when the operation '+' is performed = 100  
Result when the operation '-' is performed = 0  
Result when the operation '*' is performed = 0  
Result when the operation '/' is performed = 0  
...Program finished with exit code 0  
Press ENTER to exit console.
```

PROGRAM 9. /** Assignment: User Authentication Program

Objective

Create a C program that prompts the user for a username and password, then checks if the entered credentials match predefined values. Use logical operators to determine if the authentication is successful.

Requirements

- > Define two constants for the correct username and password.
- > Prompt the user to enter their username and password.
- > Use logical operators (&&, ||, !) to check if:
- > If both are correct, display a success message.

Implement additional checks:

- > If the username is empty, display a message indicating that the username cannot be empty.
- > If the password is empty, display a message indicating that the password cannot be empty.
- > The username matches the predefined username AND the password matches the predefined password.
- > If either the username or password is incorrect, display an appropriate error message.

*/

```
#include <stdio.h>
#include <string.h>
#define nam "Maya"
#define passwrld "password"
int main()
{
    char name[25];
    char pass[15];

    printf("Enter your name :");
    scanf("%s",name);
    if(strlen(name)==0)
    {
        printf("\nWarning : The username cannot be empty.");
    }
    printf("\nEnter your password : ");
    scanf("%s",pass);

    if(strlen(pass)==0)
    {
        printf("\nWarning : The password cannot be empty.");
    }
}
```

```

    if((strcmp(name,nam)==0) && (strcmp(pass,passwd)==0))
    {
        printf("\nAuthentication completed successfully!!!");
    }
    else{
        printf("Invalid username or password.");
    }
}

```

OUTPUT :

1) Enter your name : MAYA

Enter your password : password

Invalid username or password.

2) Enter your name : Maya

Enter your password : password

Authentication completed successfully!!!

PROGRAM 10 : /*Logic to check whether the number is even or odd .

Condition : Don't use any arithmetic operator.

PSEUDOCODE:

```

1.  N ← Read input
2.  If N & 1 == 0 then
    a.  Print "Number is even"
3.  Else
    a.  Print "Number is odd"

```

*/

```

#include <stdio.h>

```

```

int main()

```

```

{
    int a ;
    printf("Enter the number : ");
    scanf("%d",&a);
    if((a&1)==0)
    {
        printf("\nThe number is even");
    }
    else{
        printf("\nThe number is odd");
    }
}

```

```
}  
}
```

OUTPUT : 1) Enter the number : 26

The number is even

2) Enter the number : 23

The number is odd

```
PROGRAM 11 : #include <stdio.h>  
  
int main()  
{  
    int x = 2;  
    int y = ++x + x++ + --x;  
    printf("Value of y is %d.\n",y);  
  
    return 0;  
}
```

OUTPUT : Value of y is 10.

PROGRAM 12 :

```
#include <stdio.h>  
void fun(void);  
int count = 0;  
int main()  
{  
    fun();  
    fun();  
    fun();  
    return 0;  
}  
void fun()  
{  
  
    count=count+1;  
    printf("\nThe num = %d",count);  
}
```

OUTPUT :

The num = 1

The num = 2

The num = 3

PROGRAM 13 :

```
#include <stdio.h>
int main()
{
    int a= 40,b = 24;
    printf("Sum = %d",(a+b));
    printf("\nDifference = %d",(a-b));
    printf("\nProduct = %d",(a*b));
    printf("\nDivision = %d",(a/b));
    printf("\nincrement of a = %d",a++);
    printf("\nincrement of b = %d",b++);
    printf("\ndecrement of a = %d",a--);
    printf("\ndecrement of b = %d",b--);
    printf("\ndecrement of a = %d",--a);
    printf("\ndecrement of b = %d",--b);
    int c = 15;
    a=5;
    b=10;
    int result = a + b * c /b -a;
    printf("\nResult = %d",result);
    return 0;
}
```

OUTPUT :

Sum = 64

Difference = 16

Product = 960

Division = 1

increment of a = 40

increment of b = 24

decrement of a = 41

decrement of b = 25

decrement of a = 39

decrement of b = 23

Result = 15

PROGRAM 14 :

```
#include <stdio.h>
int main()
{
    char A = 10;
```

```
    printf("The value of A is : %d",A);  
  
    return 0;  
}
```

OUTPUT :

The value of A is : 10

PROGRAM 15 :

```
#include<stdio.h>  
int sum(int,int);  
int main()  
{  
    int a = 20;  
    int b =30;  
    int y = sum(a,b);  
    printf("Sum = %d",y);  
    return 0;  
}  
int sum(int c,int d)  
{  
    int sum = 0;  
    sum = c + d;  
    return sum;  
}
```

OUTPUT :

Sum = 50