**1. Arithmetic Operations on Integer**

#include <stdio.h>

int main() {

int num;

printf("Enter an integer: ");

scanf("%d", &num);

num += 10;

printf("After adding 10: %d\n", num);

num -= 5;

printf("After subtracting 5: %d\n", num);

num \*= 3;

printf("After multiplying by 3: %d\n", num);

num /= 2;

printf("After dividing by 2: %d\n", num);

num %= 7;

printf("After modulus 7: %d\n", num);

return 0;

}  
  
**Input:**

Enter an integer: 20

**Output:**

After adding 10: 30

After subtracting 5: 25

After multiplying by 3: 75

After dividing by 2: 37

After modulus 7: 2

### **2. Relational Operators**

**Question:** Write a program to demonstrate the use of relational operators. Take two integers as input and compare them using greater than, less than, equal to, and not equal to operators.

**Solution:**

#include <stdio.h>

int main() {

int a, b;

// Input two integers

printf("Enter two integers: ");

scanf("%d %d", &a, &b);

// Relational operators

printf("\nRelational operator results:\n");

// Greater than

if (a > b) {

printf("%d is greater than %d\n", a, b);

} else {

printf("%d is not greater than %d\n", a, b);

}

// Less than

if (a < b) {

printf("%d is less than %d\n", a, b);

} else {

printf("%d is not less than %d\n", a, b);

}

// Equal to

if (a == b) {

printf("%d is equal to %d\n", a, b);

} else {

printf("%d is not equal to %d\n", a, b);

}

// Not equal to

if (a != b) {

printf("%d is not equal to %d\n", a, b);

} else {

printf("%d is equal to %d\n", a, b);

}

return 0;

}

**Input:**

Enter two integers: 5 10

**Output:**

Relational operator results:

5 is not greater than 10

5 is less than 10

5 is not equal to 10

5 is not equal to 10

**3. Logical Operators**

**Question:** Write a program to demonstrate the use of logical operators (AND, OR, NOT). Use three integers and apply the logical operators.

**Solution:**

#include <stdio.h>

int main() {

int a = 5, b = 10, c = 0;

// Logical AND (&&) Operator

if (a > 0 && b > 0) {

printf("Both a and b are positive numbers.\n");

}

// Logical OR (||) Operator

if (a > 0 || c > 0) {

printf("Either a or c is a positive number.\n");

}

// Logical NOT (!) Operator

if (!(c > 0)) {

printf("c is not a positive number.\n");

}

return 0;

}

**Input:**

No input required.

**Output:**

Both a and b are positive numbers.

Either a or c is a positive number.

c is not a positive number.

**4. Bitwise Operators**

**Question:** Write a program to demonstrate bitwise operators. Perform operations like AND, OR, XOR, NOT, left shift, and right shift on two integers.

**Solution:**

#include <stdio.h>

int main() {

int a = 5, b = 9;

// Bitwise AND

printf("Bitwise AND of %d and %d is: %d\n", a, b, a & b);

// Bitwise OR

printf("Bitwise OR of %d and %d is: %d\n", a, b, a | b);

// Bitwise XOR

printf("Bitwise XOR of %d and %d is: %d\n", a, b, a ^ b);

// Bitwise NOT

printf("Bitwise NOT of %d is: %d\n", a, ~a);

// Left Shift

printf("Left Shift of %d by 1 is: %d\n", a, a << 1);

// Right Shift

printf("Right Shift of %d by 1 is: %d\n", a, a >> 1);

return 0;

}

**Input:**

No input required.

**Output:**

Bitwise AND of 5 and 9 is: 1

Bitwise OR of 5 and 9 is: 13

Bitwise XOR of 5 and 9 is: 12

Bitwise NOT of 5 is: -6

Left Shift of 5 by 1 is: 10

Right Shift of 5 by 1 is: 2

**5. Date and Time Macros**

**Question:** Write a program to display the current date and time using predefined macros \_\_DATE\_\_ and \_\_TIME\_\_.

**Solution:**

#include <stdio.h>

int main() {

// Write C code here

printf("%s\n", \_\_DATE\_\_);

printf("%s", \_\_TIME\_\_);

return 0;

}

**Input:**

No input required.

**Output:**

Nov 8 2024

10:30:45

**6. Bio Data Program**

**Question:** Write a program to collect and display the bio-data of a person, including name, age, phone number, height, weight, email, address, and educational background.

**Solution:**

#include <stdio.h>

int main() {

// Declare variables for different data types

char name[50]; // String for name

int age; // Integer for age

long phone; // Long for phone number

float height; // Float for height in meters

double weight; // Double for weight in kilograms

char email[50]; // String for email

char address[100]; // String for address

char education[100]; // String for education background

// Input biodata

printf("Enter your name: ");

scanf("%[^\n]", name); // Read string input until newline

// No need for getchar() to consume newline

printf("Enter your age: ");

scanf("%d", &age); // Read integer input

printf("Enter your phone number: ");

scanf("%ld", &phone); // Read long input

printf("Enter your height (in meters): ");

scanf("%f", &height); // Read float input

printf("Enter your weight (in kilograms): ");

scanf("%lf", &weight); // Read double input

printf("Enter your email: ");

scanf(" %[^\n]", email); // Read string input with leading space

printf("Enter your address: ");

scanf(" %[^\n]", address); // Read string input with leading space

printf("Enter your educational background: ");

scanf(" %[^\n]", education); // Read string input with leading space

// Display biodata in a resume format

printf("\n--- Resume ---\n");

printf("Name: %s\n", name);

printf("Age: %d\n", age);

printf("Phone Number: %ld\n", phone);

printf("Height: %.2f meters\n", height);

printf("Weight: %.2lf kilograms\n", weight);

printf("Email: %s\n", email);

printf("Address: %s\n", address);

printf("Educational Background: %s\n", education);

return 0; // Successful execution

}

**Input:**

Enter your name: John Doe

Enter your age: 30

Enter your phone number: 1234567890

Enter your height (in meters): 1.75

Enter your weight (in kilograms): 70.5

Enter your email: john.doe@example.com

Enter your address: 123 Main St, City, Country

Enter your educational background: Bachelors in Computer Science

**Output:**

--- Resume ---

Name: John Doe

Age: 30

Phone Number: 1234567890

Height: 1.75 meters

Weight: 70.50 kilograms

Email: john.doe@example.com

Address: 123 Main St, City, Country

Educational Background: Bachelors in Computer Science

**7. Explicit Type Casting**

**Question:** Write a program to demonstrate Explicit type casting by dividing two integers and storing the result in a double variable.

**Solution:**

#include<stdio.h>

int main() {

int a = 15, b = 2;

double c;

c = (double) a / b;

printf("%f", c);

return 0;

}

**Input:**

No input required.

**Output:**

Copy code

7.500000

**8. Implicit Type Casting (with Char)**

**Question:** Write a program to demonstrate implicit type casting between an integer and a character. Add a char value to an integer and print the result.

**Solution:**

#include<stdio.h>

int main() {

int a = 3, b;

char c = 'A';

b = a + c;

printf("%d", b);

return 0;

}

**Input:**

No input required.

**Output:**

68

**9. Simple Calculator Program**  
  
#include <stdio.h>

int main() {

char operator;

double num1, num2, result;

// Input operator from user

printf("Enter an operator (+, -, \*, /): ");

scanf(" %c", &operator);

// Input two numbers from user

printf("Enter two numbers: ");

scanf("%lf %lf", &num1, &num2);

// Perform the calculation based on the operator

if (operator == '+') {

result = num1 + num2;

printf("%.2lf + %.2lf = %.2lf\n", num1, num2, result);

}

else if (operator == '-') {

result = num1 - num2;

printf("%.2lf - %.2lf = %.2lf\n", num1, num2, result);

}

else if (operator == '\*') {

result = num1 \* num2;

printf("%.2lf \* %.2lf = %.2lf\n", num1, num2, result);

}

else if (operator == '/') {

if (num2 != 0) {

result = num1 / num2;

printf("%.2lf / %.2lf = %.2lf\n", num1, num2, result);

} else {

printf("Error! Division by zero.\n");

}

}

else {

printf("Invalid operator!\n");

}

return 0;

}

**Input:**

Enter an operator (+, -, \*, /): +

Enter two numbers: 5.5 3.2

**Output:**

5.50 + 3.20 = 8.70

**Input:**

Enter an operator (+, -, \*, /): /

Enter two numbers: 4.5 0

**Output:**

Error! Division by zero.

**10. Break**

#include <stdio.h>

int main() {

for (int i = 1; i <= 5; i++) {

if (i == 3) {

break; // Exits loop when i is 3

}

printf("%d ", i);

}

return 0;

}  
  
**Output:**

1 2

**11. Continue**

#include <stdio.h>

int main() {

for (int i = 1; i <= 5; i++) {

if (i == 3) {

continue; // Skips the rest of the loop body when i is 3

}

printf("%d ", i);

}

return 0;

}  
  
**Output:**

1 2 4 5

**12. goto**

#include <stdio.h>

int main() {

int i = 1;

start: // Label

if (i <= 5) {

printf("%d ", i);

i++;

goto start; // Jumps back to the 'start' label

}

return 0;

}  
  
**Output:**

1 2 3 4 5

**13. 2D Array Matrix in Addition**

#include <stdio.h>

int main() {

int rows, cols, i, j;

printf("Enter the number of rows: ");

scanf("%d", &rows);

printf("Enter the number of columns: ");

scanf("%d", &cols);

int matrix1[rows][cols], matrix2[rows][cols], sum[rows][cols];

// Input elements for the first matrix

printf("Enter elements of the first matrix:\n");

for (i = 0; i < rows; i++) {

for (j = 0; j < cols; j++) {

scanf("%d", &matrix1[i][j]);

}

}

// Input elements for the second matrix

printf("Enter elements of the second matrix:\n");

for (i = 0; i < rows; i++) {

for (j = 0; j < cols; j++) {

scanf("%d", &matrix2[i][j]);

}

}

// Perform matrix addition

for (i = 0; i < rows; i++) {

for (j = 0; j < cols; j++) {

sum[i][j] = matrix1[i][j] + matrix2[i][j];

}

}

// Display the result

printf("Sum of the matrices:\n");

for (i = 0; i < rows; i++) {

for (j = 0; j < cols; j++) {

printf("%d ", sum[i][j]);

}

printf("\n");

}

return 0;

}

**Iutput:**

Enter the number of rows: 2

Enter the number of columns: 3

Enter elements of the first matrix:

1 2 3

4 5 6

Enter elements of the second matrix:

6 5 4

3 2 1

**Output:**

Sum of the matrices:

7 7 7

7 7 7  
  
**14. 2D Array Matrix in Subtraction**

#include <stdio.h>

int main() {

int rows, cols, i, j;

printf("Enter the number of rows: ");

scanf("%d", &rows);

printf("Enter the number of columns: ");

scanf("%d", &cols);

int matrix1[rows][cols], matrix2[rows][cols], sum[rows][cols];

// Input elements for the first matrix

printf("Enter elements of the first matrix:\n");

for (i = 0; i < rows; i++) {

for (j = 0; j < cols; j++) {

scanf("%d", &matrix1[i][j]);

}

}

// Input elements for the second matrix

printf("Enter elements of the second matrix:\n");

for (i = 0; i < rows; i++) {

for (j = 0; j < cols; j++) {

scanf("%d", &matrix2[i][j]);

}

}

// Perform matrix addition

for (i = 0; i < rows; i++) {

for (j = 0; j < cols; j++) {

sum[i][j] = matrix1[i][j] - matrix2[i][j];

}

}

// Display the result

printf("Sum of the matrices:\n");

for (i = 0; i < rows; i++) {

for (j = 0; j < cols; j++) {

printf("%d ", sum[i][j]);

}

printf("\n");

}

return 0;

}  
  
**Input:**

Enter the number of rows: 2

Enter the number of columns: 2

Enter elements of the first matrix:

1 2

3 4

Enter elements of the second matrix:

5 6

7 8

**Output:**

Sum of the matrices:

6 8

10 12

**15. 2D Array Matrix in Transpose**

#include <stdio.h>

int main() {

int rows, cols, i, j;

// Input the number of rows and columns

printf("Enter the number of rows: ");

scanf("%d", &rows);

printf("Enter the number of columns: ");

scanf("%d", &cols);

int matrix[rows][cols], transpose[cols][rows];

// Input elements of the matrix

printf("Enter elements of the matrix:\n");

for (i = 0; i < rows; i++) {

for (j = 0; j < cols; j++) {

scanf("%d", &matrix[i][j]);

}

}

// Compute the transpose

for (i = 0; i < rows; i++) {

for (j = 0; j < cols; j++) {

transpose[j][i] = matrix[i][j];

}

}

// Display the transpose

printf("Transpose of the matrix:\n");

for (i = 0; i < cols; i++) {

for (j = 0; j < rows; j++) {

printf("%d ", transpose[i][j]);

}

printf("\n");

}

return 0;

}

**Input:**

Enter the number of rows: 2

Enter the number of columns: 3

Enter elements of the matrix:

1 2 3

4 5 6

**Output:**

Transpose of the matrix:

1 4

2 5

3 6

**16. String Comparison using strcmp()**

**Question:** Write a C program to compare two strings using the strcmp() function. The program should take two strings as input and output whether they are equal or not.

**Solution:**

#include <stdio.h>

#include <string.h>

int main() {

char str1[100], str2[100];

printf("Enter first string: ");

scanf("%s", str1);

printf("Enter second string: ");

scanf("%s", str2);

if (strcmp(str1, str2) == 0)

printf("Strings are equal.\n");

else

printf("Strings are not equal.\n");

return 0;

}

**Input:**

Enter first string: Hello

Enter second string: Hello

**Output:**

Strings are equal.

**17. String Copy using strcpy()**

**Question:** Write a C program to copy the contents of one string to another string using the strcpy() function.

**Solution:**

#include <stdio.h>

#include <string.h>

int main() {

char str1[100], str2[100];

printf("Enter source string: ");

scanf("%s", str1);

strcpy(str2, str1);

printf("Copied string: %s\n", str2);

return 0;

}

**Input:**

Enter source string: Hello

**Output:**

Copied string: Hello

**18. String Length using strlen()**

**Question:** Write a C program to find the length of a string using the strlen() function.

**Solution:**

#include <stdio.h>

#include <string.h>

int main() {

char str[100];

printf("Enter a string: ");

scanf("%s", str);

printf("Length of string: %d\n", strlen(str));

return 0;

}

**Input:**

Enter a string: Hello

**Output:**

Length of string: 5

**19. String Concatenation using strcat()**

**Question:** Write a C program to concatenate two strings using the strcat() function.

**Solution:**

#include <stdio.h>

#include <string.h>

int main() {

char str1[100], str2[100];

printf("Enter first string: ");

scanf("%s", str1);

printf("Enter second string: ");

scanf("%s", str2);

strcat(str1, str2);

printf("Concatenated string: %s\n", str1);

return 0;

}

**Input:**

Enter first string: Hello

Enter second string: World

**Output:**

Concatenated string: HelloWorld

**20. String Concatenation with Space Between Using strcat()**

**Question:** Write a C program to concatenate two strings with a space between them using strcat().

**Solution:**

#include <stdio.h>

#include <string.h>

int main() {

char str1[100], str2[100];

printf("Enter first string: ");

scanf("%s", str1);

printf("Enter second string: ");

scanf("%s", str2);

strcat(str1, " "); // Add a space

strcat(str1, str2); // Concatenate the second string

printf("Concatenated string: %s\n", str1);

return 0;

}

**Input:**

Enter first string: Hello

Enter second string: World

**Output:**

Concatenated string: Hello World

**21. Using gets and puts to read and display a string.**

#include <stdio.h>

int main() {

char str[100];

// Input a string from the user

printf("Enter a string: ");

gets(str); // Read input string

// Display the string

printf("You entered: ");

puts(str); // Output the string

return 0;

}

**Explanation**

* **gets(str);**: Reads a line of text from the user and stores it in the str array.
* **puts(str);**: Displays the stored string in str on the screen.

**Example Execution**

**Input:**

Enter a string: Hello, World!

**Output:**

You entered: Hello, World!