



replit.com/@honwadg



37



InsidiousCrushingCode ▾ ⌂ GH ▾

```
1 #include <stdio.h>
2 ~ int add(int a, int b) {
3     return a + b;
4 }
5
6 ~ int main() {
7     int num1 = 5, num2 = 10;
8     printf("Sum: %d\n", add(num1,
9         num2));
10    printf("GURURAJ");
11    return 0;
12 }
```

AI {✓} C

Ln 11, Col 2 • Spaces: 4 History ⌂

main.c



Run





replit.com/@honwadg



37



InsidiousCrushingCode ▾ ⏺

GH ▾

 Show Only Latest ⌛ Clear History

▼ Run

Ask AI

Sum: 15  
GURURAJ

&gt;\_ Console



▶ Run





replit.com/@honwadg



37



MetallicFunctionalImplementation ▾ ⌂ GH ▾

```
1 #include <stdio.h>
2 ~ int factorial(int n) {
3     if (n == 0)
4         return 1;
5     return n * factorial(n - 1);
6 }
7
8 ~ int main() {
9     int number = 5;
10    printf("Factorial of %d is %d\n",
11           number, factorial(number));
12    printf("GURURAJ");
13    return 0;
14 }
```

AI {✓} C

Ln 1, Col 1 • Spaces: 4 History ⌂

C main.c



▶ Run





replit.com/@honwadg



37



MetallicFunctionalImplementation ▾ ⌂ GH ▾

 Show Only Latest ⌛ Clear History

Run

Ask AI



Factorial of 5 is 120  
GURURAJ



&gt;\_ Console



▶ Run





ShockedCircularInformation ▾ ⌂ GH ▾

```
1 #include <stdio.h>
2 int multiply(int a, int b) {
3     return a * b;
4 }
5
6 int main() {
7     int num1 = 4, num2 = 5;
8     int result = multiply(num1, num2);
9     printf("Multiplication: %d\n",
10    result);
11    printf("GURURAJ");
12 }
```

AI {✓} C

Ln 1, Col 1 • Spaces: 2 History ⌂

C main.c



Run





replit.com/@honwadg



37



ShockedCircularInformation ▾ ⌂ GH ▾

 Show Only Latest ⌛ Clear History

Run

Ask AI



Multiplication: 20  
GURURAJ



&gt;\_ Console



▶ Run





ShockedCircularInformation ▾ ⌂ GH ▾

```
1 #include <stdio.h>
2 void checkEvenOdd(int num) {
3     if (num % 2 == 0)
4         printf("%d is Even\n", num);
5     else
6         printf("%d is Odd\n", num);
7 }
8
9 int main() {
10     int number = 7;
11     checkEvenOdd(number);
12     printf("GURURAJ");
13     return 0;
14 }
```

AI {✓} C

Ln 1, Col 1 • Spaces: 4 History ⌂

main.c



Run





replit.com/@honwadg



37



ShockedCircularInformation ▾ ⌂ GH ▾

 Show Only Latest ⌛ Clear History

▼ Run

Ask AI



7 is Odd  
GURURAJ



&gt;\_ Console



▶ Run





ImpartialDiligentNetworks

```
1 #include <stdio.h>
2 int main() {
3     int arr[5] = {10, 20, 30, 40, 50};
4     int sum = 0;
5     for (int i = 0; i < 5; i++) {
6         sum += arr[i];
7     }
8     printf("Sum of array elements: %d\n",
9            sum);
10    printf("GURURAJ");
11    return 0;
```



AI {✓} C

Ln 11, Col 2 • Spaces: 4 History

main.c



Run





replit.com/@honwadg



37



ImpartialDiligentNetworks ▾ ⌂

GH ▾

 Show Only Latest ⌛ Clear History

Run ▾

Ask AI



Sum of array elements: 150  
GURURAJ



&gt;\_ Console



▶ Run





```
ImpartialDiligentNetworks ▾ ⌂ GH ▾  
1 #include <stdio.h>  
2 √ int main() {  
3     int arr[5] = {3, 5, 7, 2, 9};  
4     int largest = arr[0];  
5  
6     for (int i = 1; i < 5; i++) {  
7         if (arr[i] > largest) {  
8             largest = arr[i];  
9         }  
10    }  
11    printf("Largest element: %d\n",  
12        largest);  
13    printf("GURURAJ");  
14    return 0;  
15 }
```

AI {✓} C

Ln 1, Col 1 • Spaces: 4 History ⌂

C main.c



Run





replit.com/@honwadg



37



ImpartialDiligentNetworks ▾ ⌂

GH ▾

 Show Only Latest ⌛ Clear History

▼ Run

Ask AI



Largest element: 9  
GURURAJ



&gt;\_ Console



▶ Run





ImpartialDiligentNetworks ▾ ⌂ GH ▾

```
1 #include <stdio.h>
2 √ int main() {
3     int arr[5] = {1, 2, 3, 4, 5};
4     printf("Original array: ");
5     √ for (int i = 0; i < 5; i++) {
6         printf("%d ", arr[i]);
7     }
8     printf("\nReversed array: ");
9     √ for (int i = 4; i >= 0; i--) {
10        printf("%d ", arr[i]);
11    }
12    printf("\n");
13    printf("GURURAJ");
14    return 0;
15 }
```

AI {✓} C

Ln 1, Col 1 • Spaces: 4 History ⌂

C main.c



▶ Run





replit.com/@honwadg



37



ImpartialDiligentNetworks ▾ ⌂

GH ▾

 Show Only Latest ⌛ Clear History

▼ Run

Ask AI



Original array: 1 2 3 4 5

Reversed array: 5 4 3 2 1

GURURAJ



&gt;\_ Console



▶ Run





```
ImpartialDiligentNetworks ▾ ⌂ GH ▾
1 #include <stdio.h>
2 √ int main() {
3     int arr[6] = {2, 7, 4, 9, 6, 5};
4     int evenCount = 0, oddCount = 0;
5     for (int i = 0; i < 6; i++) {
6         if (arr[i] % 2 == 0)
7             evenCount++;
8         else
9             oddCount++;
10    }
11    printf("Even numbers: %d\n",
evenCount);
12    printf("Odd numbers: %d\n",
oddCount);
13    printf("GURURAJ");
14    return 0;
15 }
```

AI {✓} C

Ln 1, Col 1 • Spaces: 4 History ⌂

C main.c



▶ Run





replit.com/@honwadg



37



ImpartialDiligentNetworks ▾ ⌂

GH ▾

 Show Only Latest ⌛ Clear History

▼ Run

Ask AI



Even numbers: 3

Odd numbers: 3

GURURAJ



&gt;\_ Console



▶ Run





```
ImpartialDiligentNetworks ▾ ⌂ GH ▾
1 #include <stdio.h>
2 √ int main() {
3     int num = 10;
4     int *ptr = &num;
5
6     printf("Value of num: %d\n", num);
7     printf("Address of num: %p\n",
&num);
8     printf("Value of ptr: %p\n", ptr);
9     printf("Value pointed by ptr:
%d\n", *ptr);
10    printf("GURURAJ");
11    return 0;
12 }
```

AI {✓} C

Ln 1, Col 1 • Spaces: 4 History ⌂

C main.c



▶ Run





replit.com/@honwadg



37



ImpartialDiligentNetworks ▾ ⌂

GH ▾

 Show Only Latest ⌛ Clear History

▼ Run

Ask AI



```
Value of num: 10
Address of num: 0x7ffcdd6f7964
Value of ptr: 0x7ffcdd6f7964
Value pointed by ptr: 10
GURURAJ
```



&gt;\_ Console



▶ Run





ImpartialDiligentNetworks ▾ Cloud GH ▾

```
1 #include <stdio.h>
2 void swap(int *a, int *b) {
3     int temp = *a;
4     *a = *b;
5     *b = temp;
6 }
7 int main() {
8     int x = 5, y = 10;
9     printf("Before Swap: x = %d, y =
%d\n", x, y);
10    swap(&x, &y);
11    printf("After Swap: x = %d, y =
%d\n", x, y);
12    printf("GURURAJ");
13    return 0;
14 }
```

AI {✓} C

Ln 1, Col 1 • Spaces: 4 History ⌂

C main.c



Run





replit.com/@honwadg



37



ImpartialDiligentNetworks ▾ ⌂

GH ▾

 Show Only Latest ⌛ Clear History

▼ Run

Ask AI



Before Swap: x = 5, y = 10  
After Swap: x = 10, y = 5  
GURURAJ



&gt;\_ Console



▶ Run





```
ImpartialDiligentNetworks ▾ ⌂ GH ▾  
1 #include <stdio.h>  
2 √ int main() {  
3     int arr[5] = {10, 20, 30, 40, 50};  
4     int *ptr = arr;  
5  
6     printf("Array elements: ");  
7     for (int i = 0; i < 5; i++) {  
8         printf("%d ", *(ptr + i));  
9     }  
10    printf("\n");  
11    printf("GURURAJ");  
12    return 0;  
13 }
```

AI {✓} C

Ln 1, Col 1 • Spaces: 4 History ⌂

C main.c



▶ Run





replit.com/@honwadg



37



ImpartialDiligentNetworks ▾ ⌂

GH ▾

 Show Only Latest ⌛ Clear History

Run

Ask AI



Array elements: 10 20 30 40 50  
GURURAJ

&gt;\_ Console



Run





ImpartialDiligentNetworks ▾ ⌂ GH ▾

```
1 #include <stdio.h>
2 √ int add(int a, int b) {
3     return a + b;
4 }
5 √ int main() {
6     int (*funcPtr)(int, int) = add;
7
8     int result = funcPtr(10, 20);
9     printf("Sum: %d\n", result);
10    printf("GURURAJ");
11    return 0;
12 }
```



AI {✓} C

Ln 12, Col 2 • Spaces: 4 History ⌂

main.c



Run





replit.com/@honwadg



37



ImpartialDiligentNetworks ▾ ⌂

GH ▾

 Show Only Latest ⌛ Clear History

▼ Run

Ask AI

Sum: 30  
GURURAJ

&gt;\_ Console



▶ Run





ImpartialDiligentNetworks ▾ ⌂ GH ▾

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 √ int main() {
4     int n, *arr, sum = 0;
5     printf("Enter number of elements:
"); 
6     scanf("%d", &n);
7     arr = (int *)malloc(n *
sizeof(int));
8 √     if (arr == NULL) {
9         printf("Memory allocation
failed\n");
10        return 1;
11    }
12    printf("Enter %d elements:\n", n);
13 √    for (int i = 0; i < n; i++) {
14        scanf("%d", &arr[i]);
15        sum += arr[i];
16    }
17    printf("Sum of elements: %d\n",
sum);
18    free(arr);
19    printf("GURURAJ");
20    return 0;
21 }
```

AI {✓} C

Ln 21, Col 2 • Spaces: 4 History ⌂

C main.c



▶ Run





replit.com/@honwadg



37



ImpartialDiligentNetworks ▾ ⌂

GH ▾

 Show Only Latest ⌛ Clear History

Run

Ask AI



Enter number of elements: 2 2

Enter 2 elements:

2 3

Sum of elements: 4

GURURAJ



&gt;\_ Console



▶ Run





ImpartialDiligentNetworks ▾ ⌂ GH ▾

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 √ int main() {
5     int n, *arr;
6     printf("Enter number of elements:
7     ");
8     scanf("%d", &n);
9     arr = (int *)calloc(n,
10         sizeof(int));
11    if (arr == NULL) {
12        printf("Memory allocation
13 failed\n");
14        return 1;
15    }
16    for (int i = 0; i < n; i++) {
17        printf("%d ", arr[i]);
18    }
19    printf("\n");
20    free(arr);
21    printf("GURURAJ");
22    return 0;
23 }
```

AI {✓} C

Ln 1, Col 1 • Spaces: 4 History ⌂

C main.c



▶ Run





replit.com/@honwadg



37



ImpartialDiligentNetworks ▾ ⌂

GH ▾

 Show Only Latest ⌛ Clear History

Run ▾

Ask AI



```
Enter number of elements: 2 3
Array elements initialized to 0:
0 0
GURURAJ
```



&gt;\_ Console



▶ Run





ImpartialDiligentNetworks ▾ ⌂ GH ▾

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 int main() {
4     int n, new_size, *arr;
5     printf("Enter number of elements:
6     ");
7     scanf("%d", &n);
8     arr = (int *)malloc(n *
9         sizeof(int));
10    if (arr == NULL) {
11        printf("Memory allocation
12 failed\n");
13        return 1;
14    }
15    printf("Enter %d elements:\n", n);
16    for (int i = 0; i < n; i++) {
17        scanf("%d", &arr[i]);
18    }
19    printf("Enter new size of the
20 array: ");
21    scanf("%d", &new_size);
22    arr = (int *)realloc(arr,
23        new_size * sizeof(int));
24    if (arr == NULL) {
25        printf("Memory reallocation
26 failed\n");
27        return 1;
28    }
```

AI {✓} C

Ln 1, Col 1 • Spaces: 4 History ⌂

C main.c



Run





ImpartialDiligentNetworks ▾ ⌂ GH ▾

```
20         printf("Memory reallocation
failed\n");
21         return 1;
22     }
23     if (new_size > n) {
24         printf("Enter additional %d
elements:\n", new_size - n);
25     for (int i = n; i < new_size;
26         i++) {
27         scanf("%d", &arr[i]);
28     }
29     printf("Updated array
elements:\n");
30     for (int i = 0; i < new_size;
31         i++) {
32         printf("%d ", arr[i]);
33     printf("\n");
34     free(arr);
35     printf("GURURAJ");
36     return 0;
37 }
```

AI { } C

Ln 1, Col 1 • Spaces: 4 History ⌂

C main.c



Run





replit.com/@honwadg



37



ImpartialDiligentNetworks ▾ ⌂

GH ▾

 Show Only Latest ⌕ Clear History

Run

Ask AI



Enter number of elements: 2 3

Enter 2 elements:

4 5

Enter new size of the array: Enter additional  
l 3 elements:

1 2 3

Updated array elements:

3 4 1 2 3

GURURAJ



&gt;\_ Console



▶ Run





DeeppinkDeafeningStartups ▾ ⌂ GH ▾

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 int main() {
4     int n, *arr;
5     printf("Enter number of elements:
");  
6     scanf("%d", &n);
7     arr = (int *)malloc(n *
sizeof(int));
8     if (arr == NULL) {
9         printf("Memory allocation
failed\n");
10        return 1;
11    }
12    printf("Enter %d elements:\n", n);
13    for (int i = 0; i < n; i++) {
14        scanf("%d", &arr[i]);
15    }
16    printf("Stored elements are:\n");
17    for (int i = 0; i < n; i++) {
18        printf("%d ", arr[i]);
19    }
20    printf("\n");
21    free(arr);
22    printf("Memory has been
freed.\n");
23    printf("GURURAJ");
24    return 0.
```

🕒 AI {✓} C

Ln 23, Col 23 • Spaces: 4 History ⌂

C main.c



▶ Run





replit.com/@honwadg



37



DeeppinkDeafeningStartups ▾

GH ▾

 Show Only Latest  Clear History

Run

Ask AI



Enter number of elements: 4 3

Enter 4 elements:

4 3 2 1

Stored elements are:

3 4 3 2

Memory has been freed.

GURURAJ



&gt;\_ Console



Run

