

1. Program on Operator in C

1.Arithmetic Operators

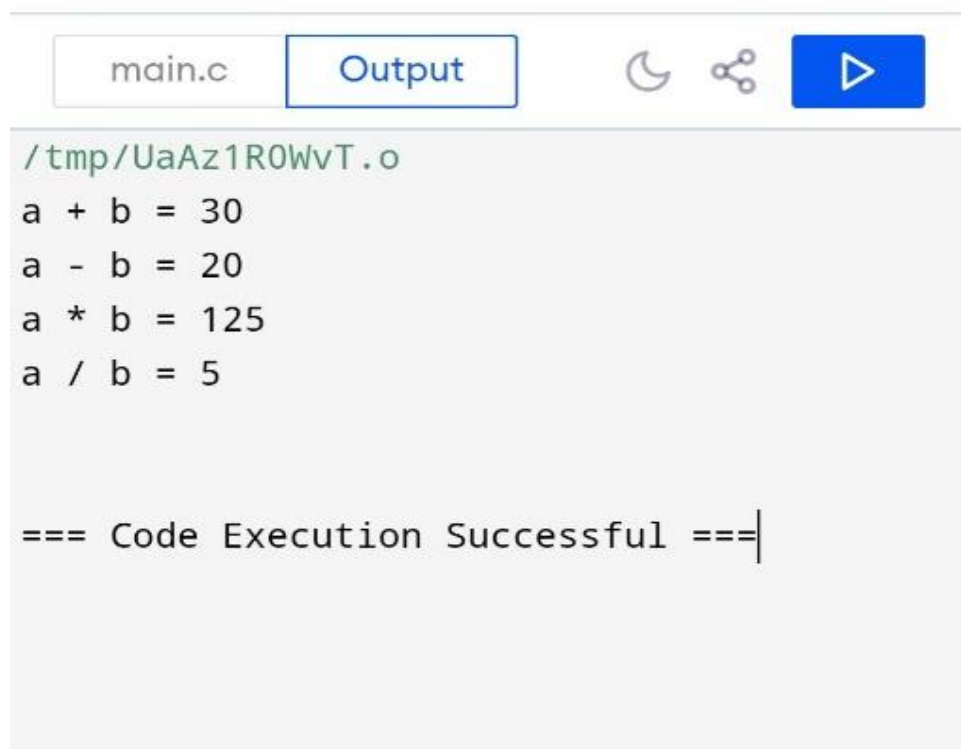
*Program

```
#include <stdio.h>

int main()
{
    int a = 25, b = 5;

    printf("a + b = %d\n", a + b);
    printf("a - b = %d\n", a - b);
    printf("a * b = %d\n", a * b);
    printf("a / b = %d\n", a / b);
    return 0;
}
```

*Output

A screenshot of a code editor or IDE showing the output of a C program. At the top, there are tabs for 'main.c' and 'Output', with 'Output' being the active tab. To the right of the tabs are icons for a moon (theme toggle), a share icon, and a blue play button. Below the tabs, the output text is displayed in a monospaced font. It starts with a green path '/tmp/UaAz1R0WvT.o' followed by four lines of arithmetic results: 'a + b = 30', 'a - b = 20', 'a * b = 125', and 'a / b = 5'. At the bottom, a status message reads '=== Code Execution Successful ===' with a cursor at the end.

```
main.c  Output  [moon] [share] [play]

/tmp/UaAz1R0WvT.o
a + b = 30
a - b = 20
a * b = 125
a / b = 5

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

2.Logical Operator

*Program

```
#include <stdio.h>
int main()
{
    int a = 5, b = 5, c = 10, result;

    result = (a == b) && (c > b);
    printf("(a == b) && (c > b) is %d \n", result);




    result = (a == b) || (c < b);
    printf("(a == b) || (c < b) is %d \n", result);

    return 0;
}
```

*Output

main.c

Output

```
/tmp/kqrLDSayCq.o
(a == b) && (c > b) is 1
(a == b) || (c < b) is 1

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

3.Relational Operator

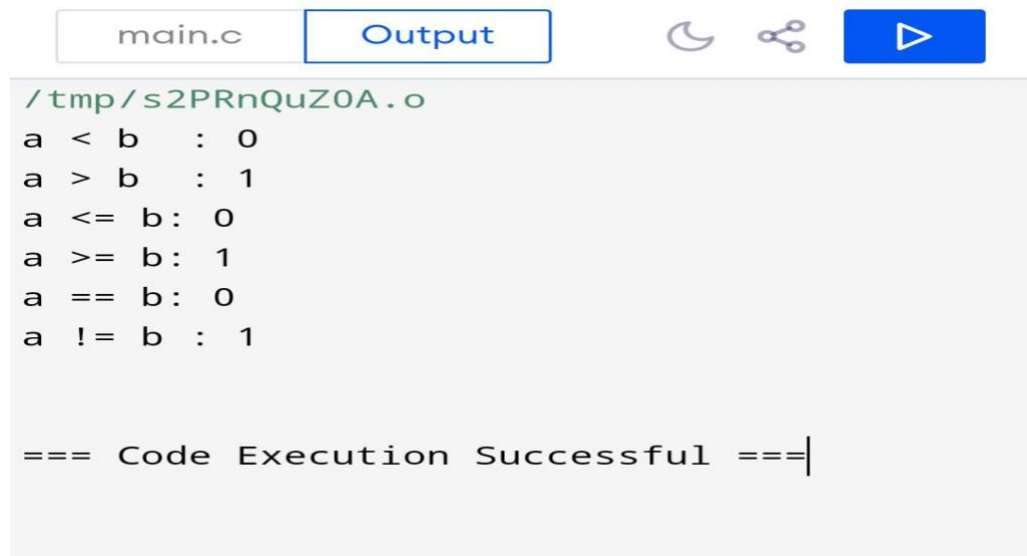
*Program

```
#include <stdio.h>

int main()
{
    int a = 25, b = 5;
    printf("a < b : %d\n", a < b);
    printf("a > b : %d\n", a > b);
    printf("a <= b: %d\n", a <= b);
    printf("a >= b: %d\n", a >= b);
    printf("a == b: %d\n", a == b);
    printf("a != b : %d\n", a != b);

    return 0;
}
```

*Output



```
main.c  Output  [Icons: Moon, Share, Play]

/tmp/s2PRnQuZ0A.o
a < b    : 0
a > b    : 1
a <= b   : 0
a >= b   : 1
a == b   : 0
a != b   : 1

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

4.Bitwise Operator

*Program

```
#include <stdio.h>

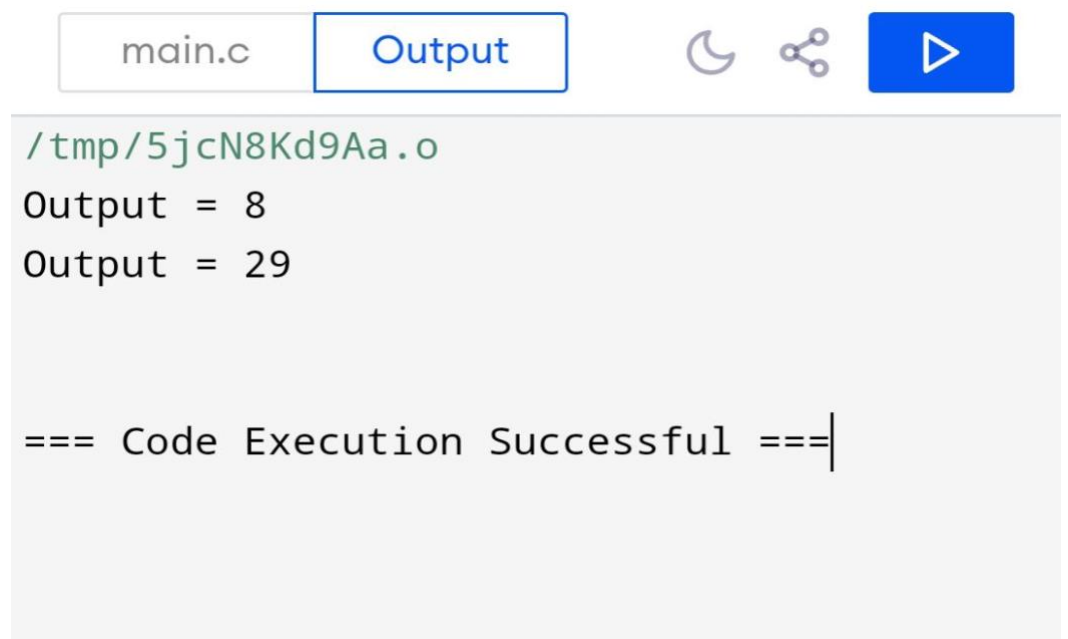
int main() {

    int a = 12, b = 25;
    printf("Output = %d\n", a & b);

    printf ("Output = %d\n", a | b);

    return 0;
}
```

*Output



```
/tmp/5jcN8Kd9Aa.o
Output = 8
Output = 29

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

5. Conditional Operators

*Program

```
#include <stdio.h>

int main()
{

    int a = 21;
    int c ;

    c = a;
    printf("output c=a = %d\n", c );

    c+= a;
    printf("Output c+=a = %d\n", c );

    c -= a;
    printf("Output c-=a = %d\n", c );

    c *= a;
    printf(" Output c*=a = %d\n", c );

    c /= a;
    printf("Output c/=a = %d\n", c );
    return 0;
}
```

*Output

main.c

Output



```
/tmp/GaGnuOCEpY.o
```

```
output c=a = 21
```

```
Output c+=a = 42
```

```
Output c-=a = 21
```

```
    Output c*=a = 441
```

```
Output c/=a  = 21
```

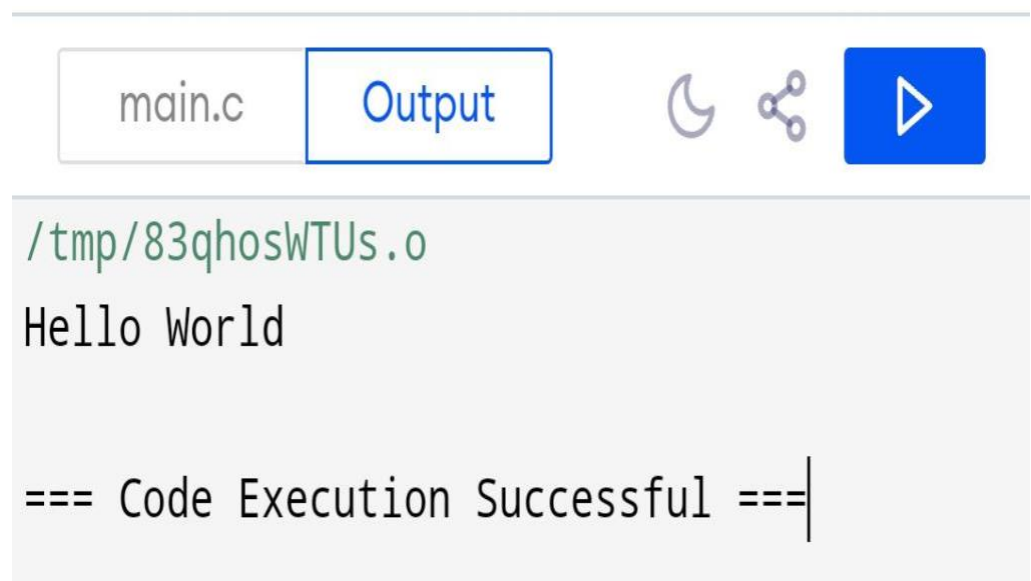
```
=== Code Execution Successful ===|
```

- Hello world Program

```
#include <stdio.h>
```

```
int main ()  
{  
    print of (" Hello world");  
}  
return 0;
```

*Output



The screenshot shows a code editor interface. At the top, there are two tabs: 'main.c' and 'Output'. The 'Output' tab is active. To the right of the tabs are three icons: a moon (dark mode), a share icon, and a blue play button. Below the tabs, the output of the program is displayed in a light gray box. It shows the file path '/tmp/83qhosWTUs.o', the output 'Hello World', and a success message '=== Code Execution Successful ===' with a cursor at the end.

```
main.c  Output  [moon] [share] [play]  
  
/tmp/83qhosWTUs.o  
Hello World  
  
=== Code Execution Successful ===|
```