



1. Bestimmen Sie, welche Werte in den Variablen enthalten sind!
2. Überprüfen Sie die Lösung mit einem Programm!

```
int a = 5;  
int b = 3;  
int c = 5/3; ergibt ?    1
```

```
int a = 5;  
int b = 4;  
int c = 8;  
int d = a+b *c: ergibt?  72
```

```
int a = 5;  
bool k = false;  
bool m = (a > 4); ergibt ?    true  
bool n = (a == 4); ergibt ?   false
```

```
unsigned char a = 7;  
int b = 100; int c = a * b / b; // ergibt ?    7  
int d = 10000; int e = a * d / d; // ergibt ?   7
```

```
signed char a = 7;  
signed char b = 100;  
int c = a * b; // ergibt ?    700  
int d = 700; int e = 100;  
long f = d * e; // ergibt ?   70.000
```

```
int a = 5;  
int b = 4;
```

```
int c = 8;
int d = a+b *c: ergibt? 72
```

```
int a = (13/4) % 2; ergibt ? 3,90
```

```
int a = ((13/4) % 2) + (((5/4)*3 / 7);
```

Liste der Operatoren nach Priorität:

[https://de.wikibooks.org/wiki/C-Programmierung:\\_Liste\\_der\\_Operatoren\\_nach\\_Priorit%C3%A4t](https://de.wikibooks.org/wiki/C-Programmierung:_Liste_der_Operatoren_nach_Priorit%C3%A4t)

```
int a = 1;
bool b = false;
```

```
bool c = (a + a *5 % 4 > a/ 4 && ! b || a ==1); ergibt? 1 a*5 dann 5%4 dann a+(5%4) dann /4 dann !b || a ==1
ergebnis: -> a == 1 = true
```

Reihenfolge:

Von links nach rechts!

NICHT !

Punktrechnung \*

Division /

Modulo %

Vergleich größer >

Vergleich gleich ==

Oder ||

Und &&

```
char a = 0x3A;
char b = 0x76;
char c = ~a;
char d = a | b; ergibt ? ~
char e = a & b; ergibt ? 2
char f = a ^ b; ergibt ? L
```