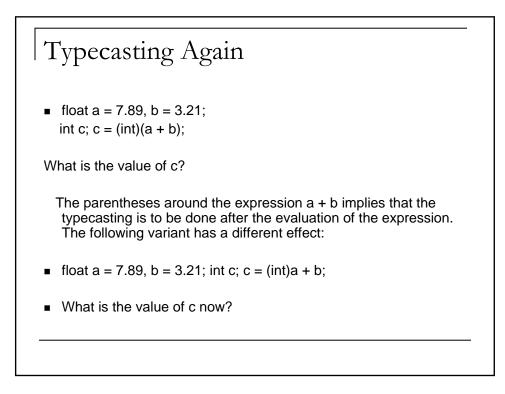


Assignments

## Example

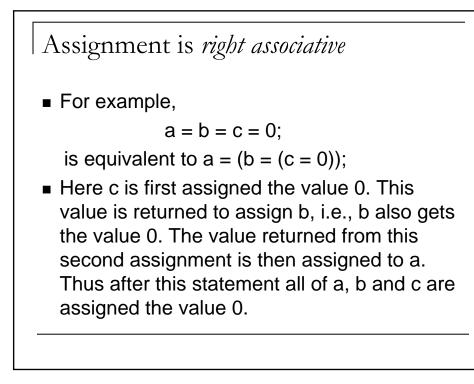
```
#include<stdio.h>
main()
{
  float a = -7.89., b = 3;
  int c;
  typedef unsigned long newlong;
  newlong d;
  c = (int) a + b;
  d=c;
  printf("%d\n",c);
  printf("%x\n",d);
}
```

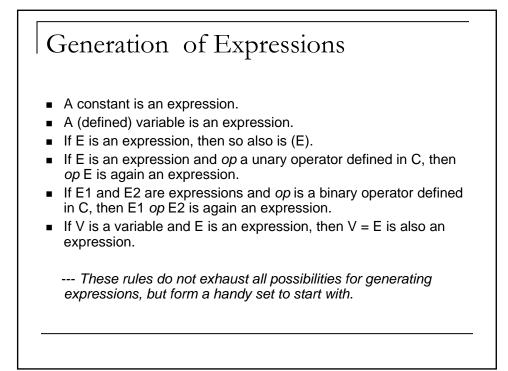


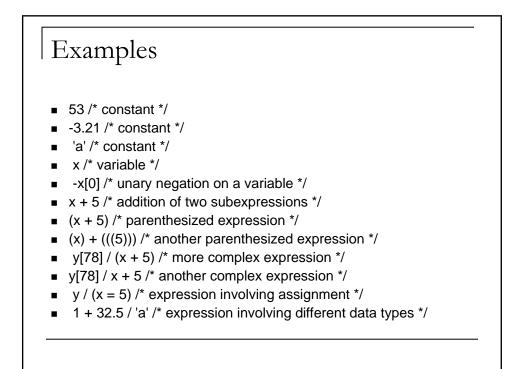
Assignments also return a value.

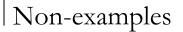
■ int a, b, c; c = (a = 8) + (b = 13);

- Here a is assigned the value 8 and b the value 13. The values (8 and 13) returned by these assignments are then added and the sum 21 is stored in c.
- The assignment of c also returns a value, i.e., 21.
- Here we do not need this value.

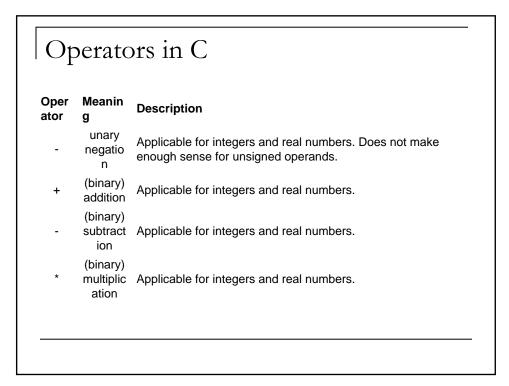




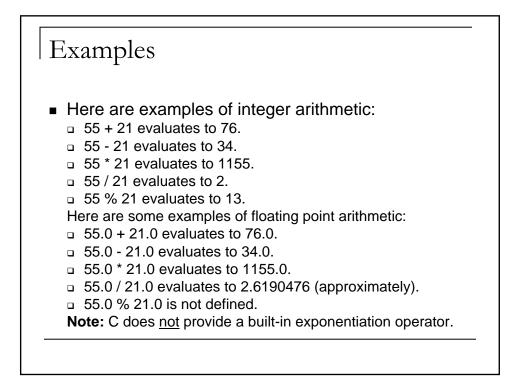


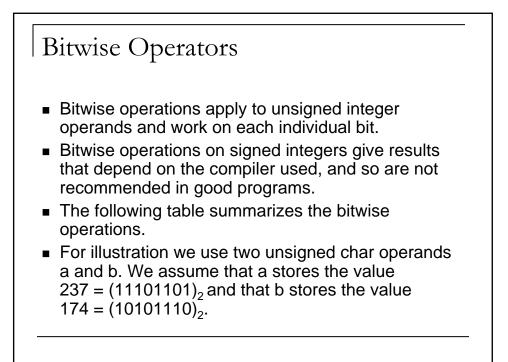


- 5 3 /\* space is not an operator and integer constants may not contain spaces \*/
- y \*+ 5 /\* \*+ is not a defined operator \*/
- x (+ 5) /\* badly placed parentheses \*/
- x = 5; /\* semi-colons are not allowed in expressions \*/



	Op	erato	rs in C
-	/	(binary) division	For integers division means "quotient", whereas for real numbers division means "real division". If both the operands are integers, the integer quotient is calculated, whereas if (one or both) the operands are real numbers, real division is carried out.
	%	(binary) remaind er	Applicable only for integer operands.
_			



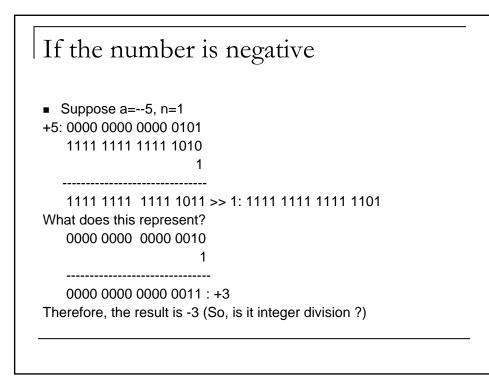


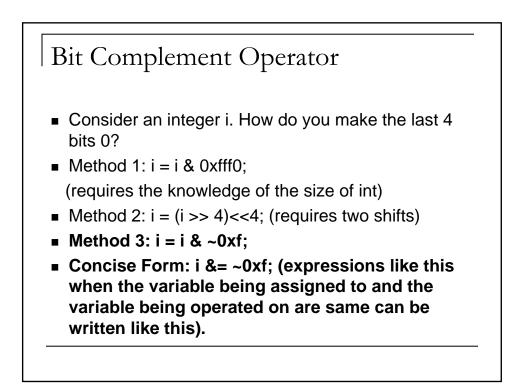
Operator	Meaning	Example										
&	AND	a = 237	1	1	1	0	1	1	0	1		
		b = 174	1	0	1	0	1	1	1	0		
		a & b is 172	1	0	1	0	1	1	0	0		
		a = 237	1	1	1	0	1	1	0	1		
Ι	OR	b = 174	1	0	1	0	1	1	1	0		
		a   b is 239	1	1	1	0	1	1	1	1		
	EXOR	a = 237	1	1	1	0	1	1	0	1		
^		b = 174	1	0	1	0	1	1	1	0		
		a ^ b is 67	0	1	0	0	0	0	1	1		
	Complement	a = 237	1	1	1	0	1	1	0	1		
~	Complement	~a is 18	0	0	0	1	0	0	1	0	_	
	Dight chift	a = 237	1	1	1	0	1	1	0	1		
>>	Right-shift	a >> 2 is 59	0	1	1	1	0	1	1	0		
	Left-shift	b = 174	1	0	1	0	1	1	1	0		
<<		b << 1 is 92	0	1	0	1	1	1	0	0		

```
Multiply by 2 (or powers of 2)
```

```
#include<stdio.h>
main()
{
    int a;
    int n;
        scanf("%d",&a);
        scanf("%d",&n);
        printf("Result: %d\n",a<<n);
}</pre>
```

```
Divide by 2 (or powers of 2)
#include<stdio.h>
main()
{
    int a;
    int n;
    scanf("%d",&a);
    scanf("%d",&n);
    printf("Result: %d\n",a>>n);
}
```





```
Extract the n<sup>th</sup> bit

#include<stdio.h>
main()
{
    int i, n;
    int bit;
    scanf("%d",&i);
    scanf("%d",&n);
    bit = (i>>n)&1;
    printf("The %dth bit of %d is %d\n",n,i,bit);
}
```

