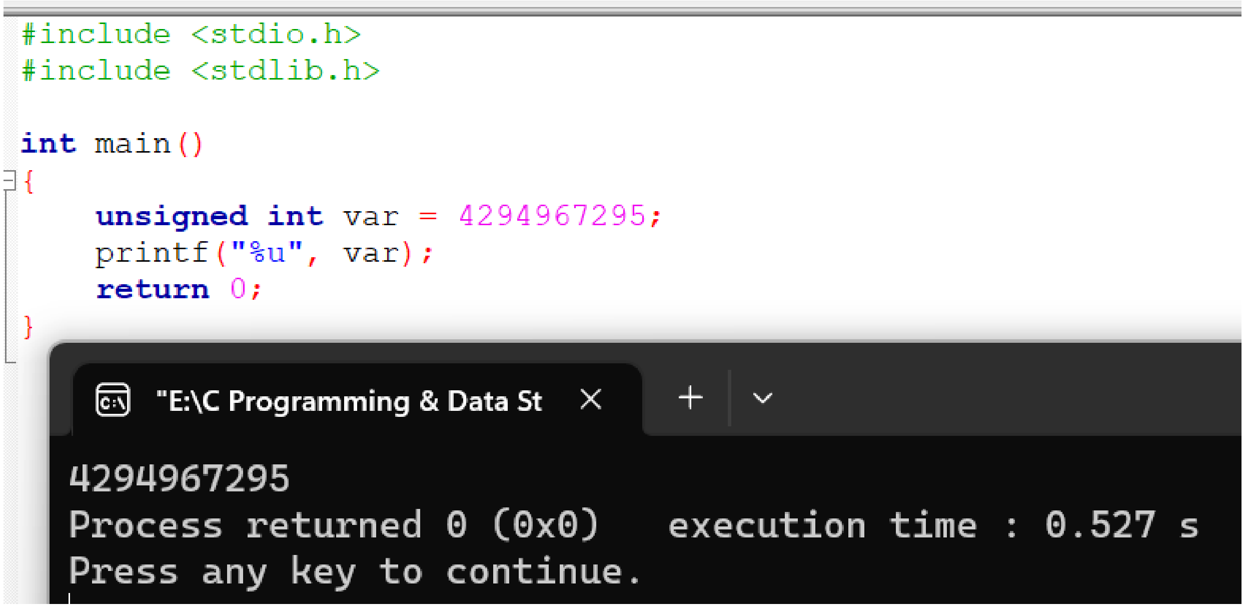
Lesson 8. Exceeding The Valid Range of Data Types

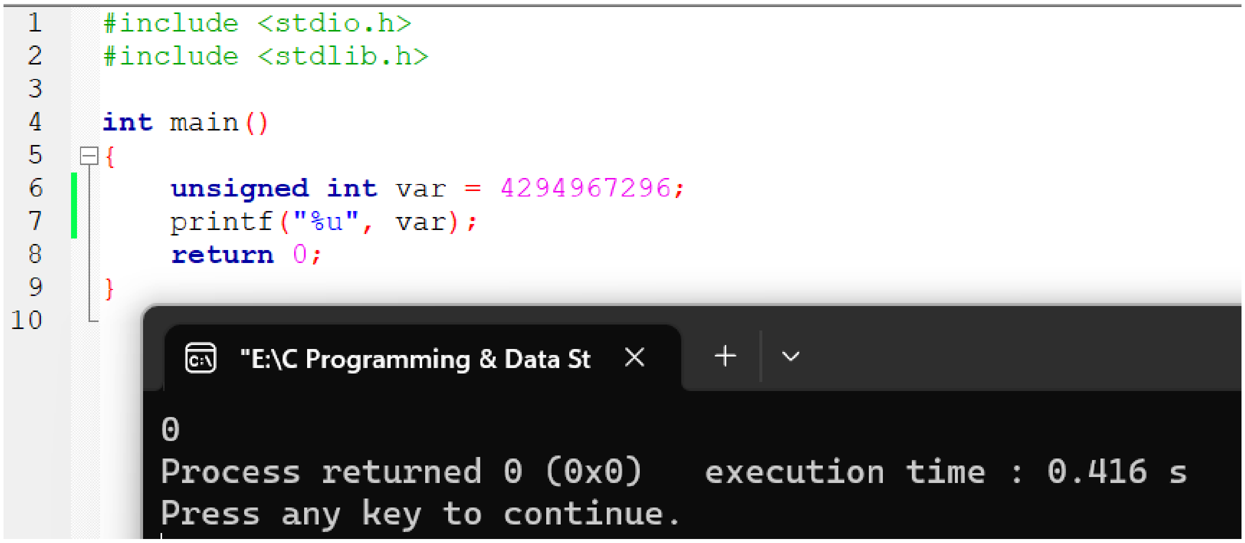
Exceeding the unsigned range

```c
#include <stdio.h>

int main(){
    unsigned int var = 4294967295;   //由lesson_7可知4294967295是无符号最大允许值
    printf("%u", var);
    return 0;
}
```



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

int main()
{
    unsigned int var = 4294967295;
    printf("%u", var);
    return 0;
}
```

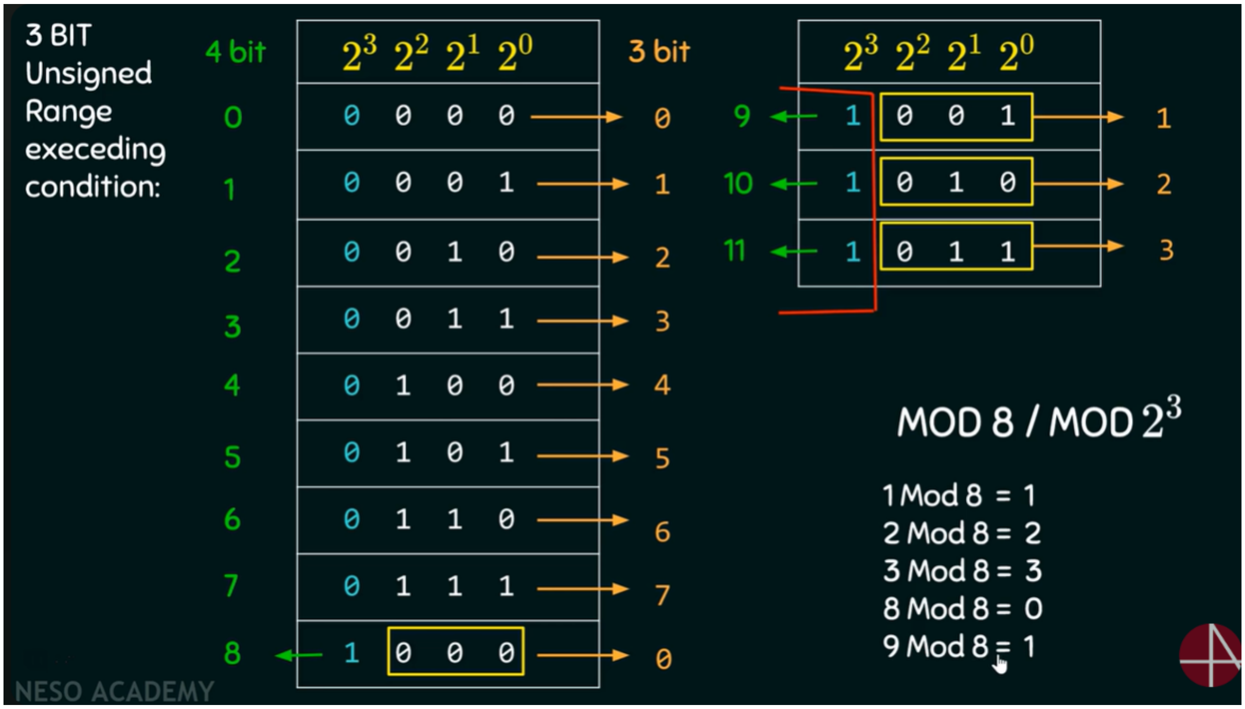
```
4294967295
Process returned 0 (0x0)   execution time : 0.527 s
Press any key to continue.
```



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

int main()
{
    unsigned int var = 4294967296;
    printf("%u", var);
    return 0;
}
```

```
0
Process returned 0 (0x0)   execution time : 0.416 s
Press any key to continue.
```

Why? (Such as the following example)



3 BIT Unsigned Range execeding condition:

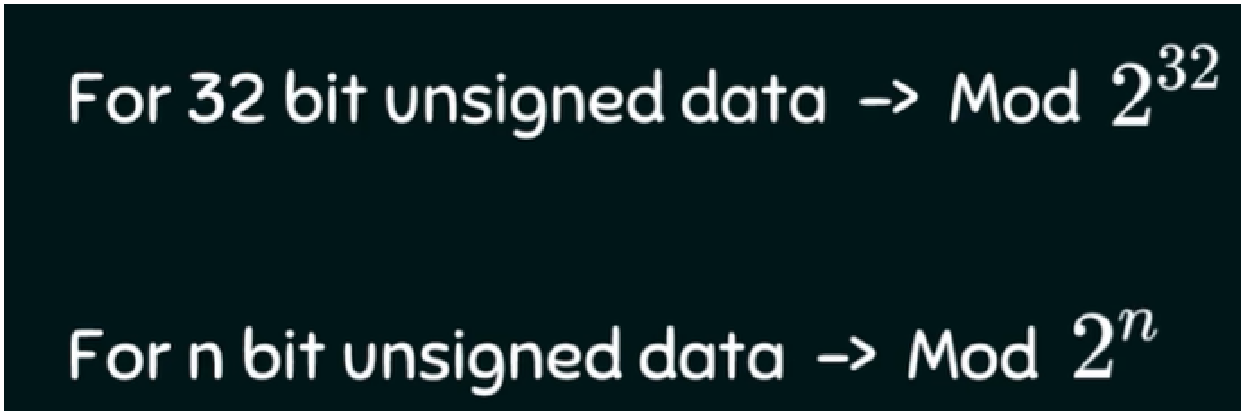
4 bit	2^3	2^2	2^1	2^0	3 bit
0	0	0	0	0	0
1	0	0	0	1	1
2	0	0	1	0	2
3	0	0	1	1	3
4	0	1	0	0	4
5	0	1	0	1	5
6	0	1	1	0	6
7	0	1	1	1	7
8	1	0	0	0	0

	2^3	2^2	2^1	2^0	
9	1	0	0	1	1
10	1	0	1	0	2
11	1	0	1	1	3

MOD 8 / MOD 2^3

1 Mod 8 = 1
2 Mod 8 = 2
3 Mod 8 = 3
8 Mod 8 = 0
9 Mod 8 = 1

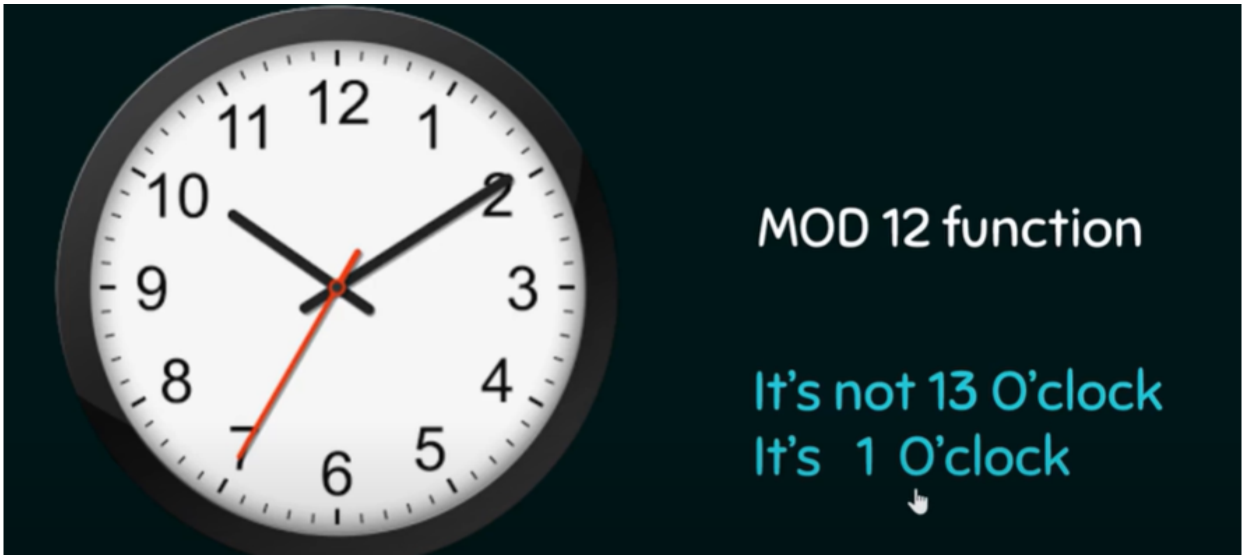
NESO ACADEMY



For 32 bit unsigned data -> Mod 2^{32}

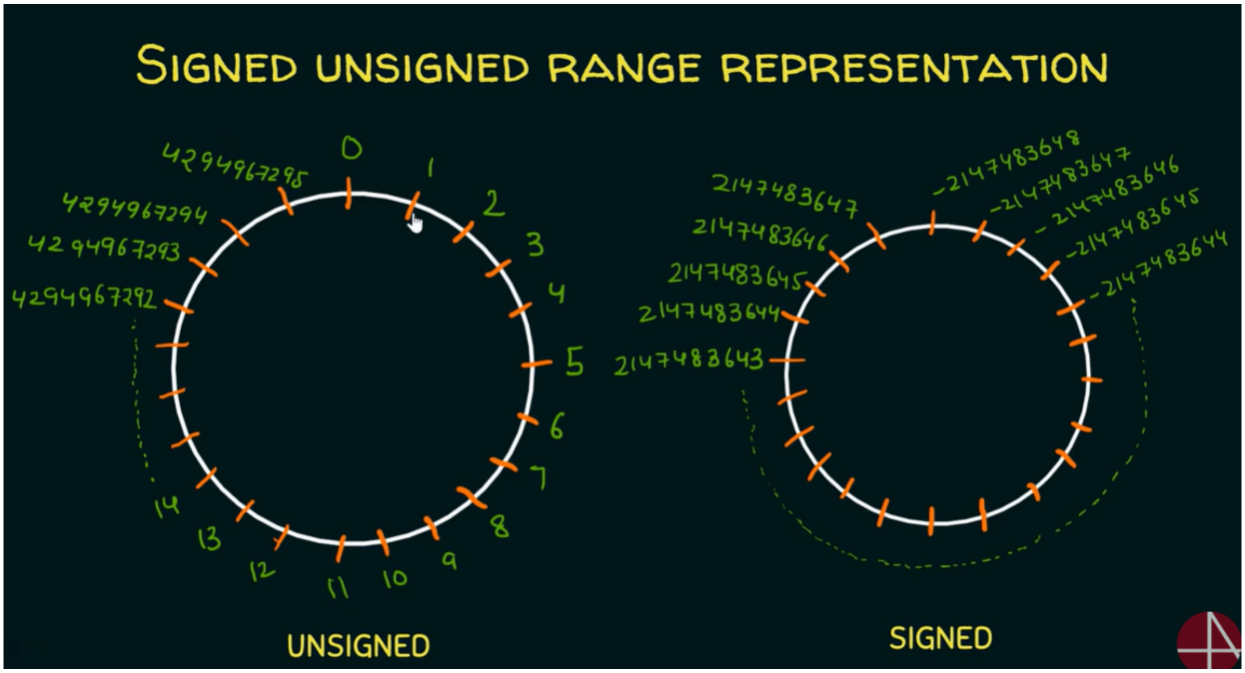
For n bit unsigned data -> Mod 2^n

the analogy

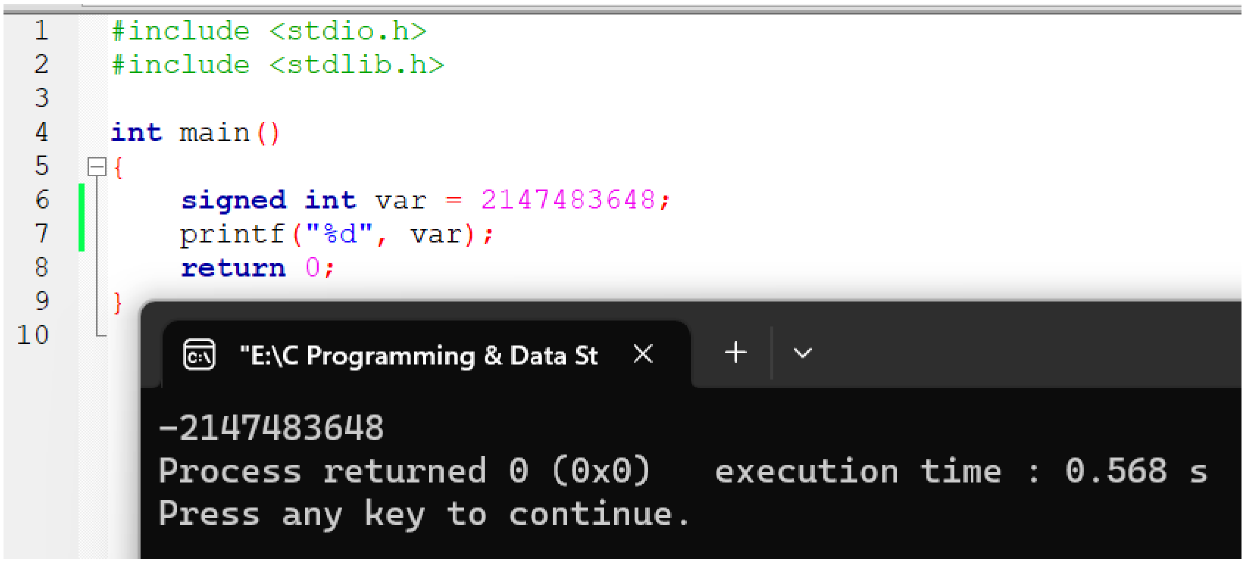


MOD 12 function

It's not 13 O'clock
It's 1 O'clock

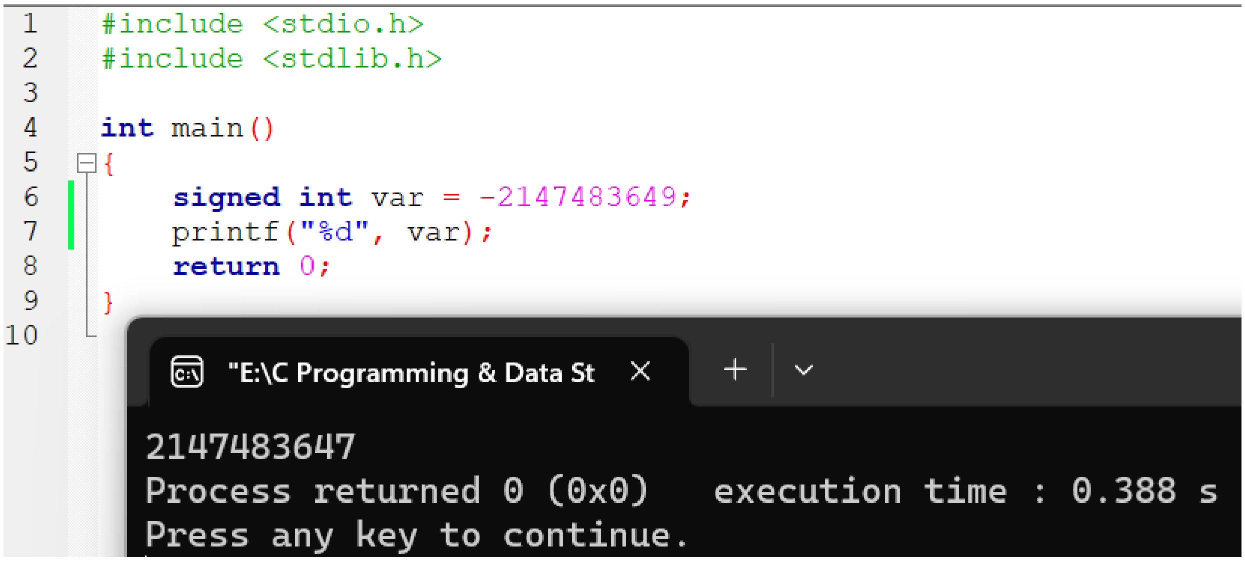


Exceeding the signed range



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

int main()
{
    signed int var = 2147483648;
    printf("%d", var);
    return 0;
}
```

```
-2147483648
Process returned 0 (0x0)   execution time : 0.568 s
Press any key to continue.
```



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

int main()
{
    signed int var = -2147483649;
    printf("%d", var);
    return 0;
}
```

```
2147483647
Process returned 0 (0x0)   execution time : 0.388 s
Press any key to continue.
```