REVIEW

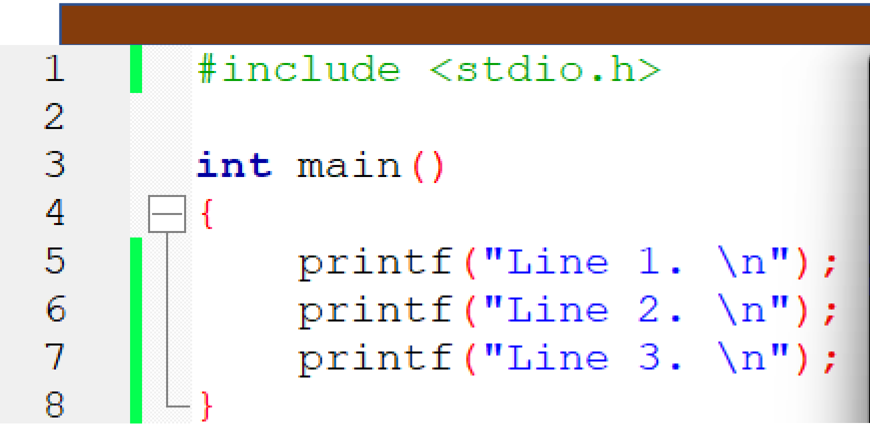
1)COMMENTING

2)PRINTF & PUTS

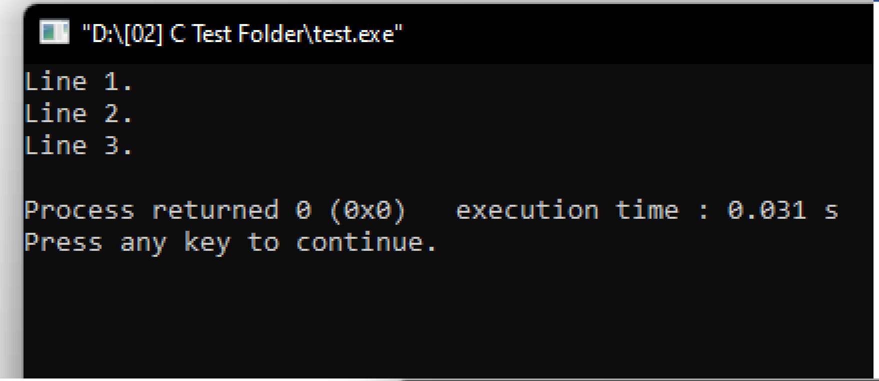
3)DATA TYPES

4)BUILT-IN FUNCTIONS

COMMENTING

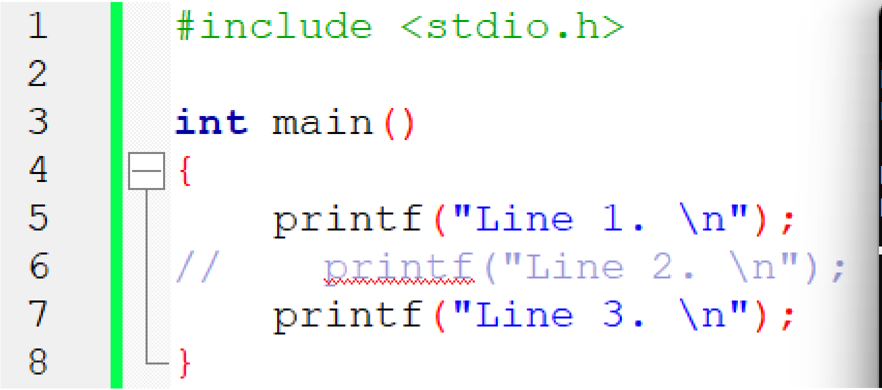


```c
#include <stdio.h>

int main()
{
    printf("Line 1. \n");
    printf("Line 2. \n");
    printf("Line 3. \n");
}
```

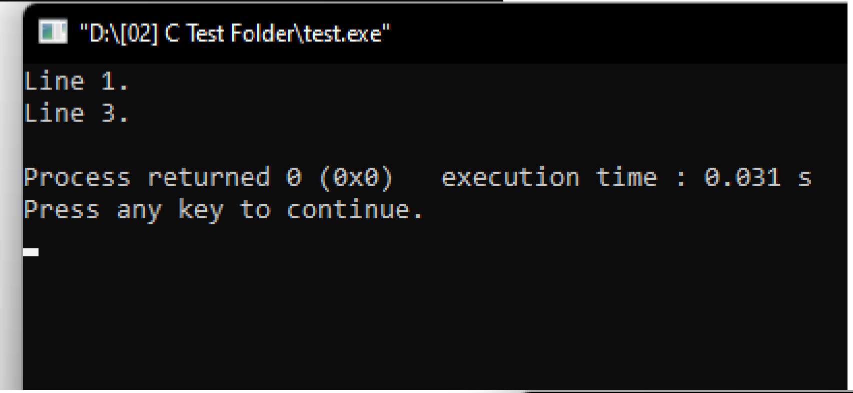


```
Line 1.
Line 2.
Line 3.

Process returned 0 (0x0)   execution time : 0.031 s
Press any key to continue.
```

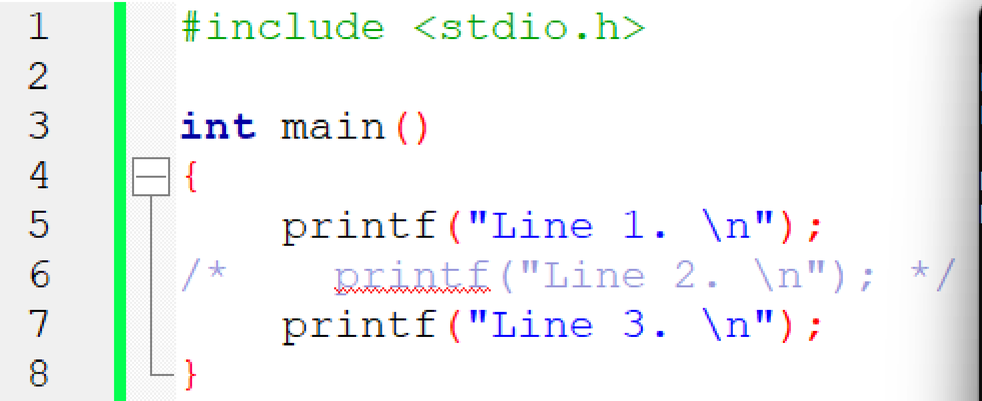
Ctrl + Shift + C: Comment a block
Ctrl + Shift + X: Un-comment a block



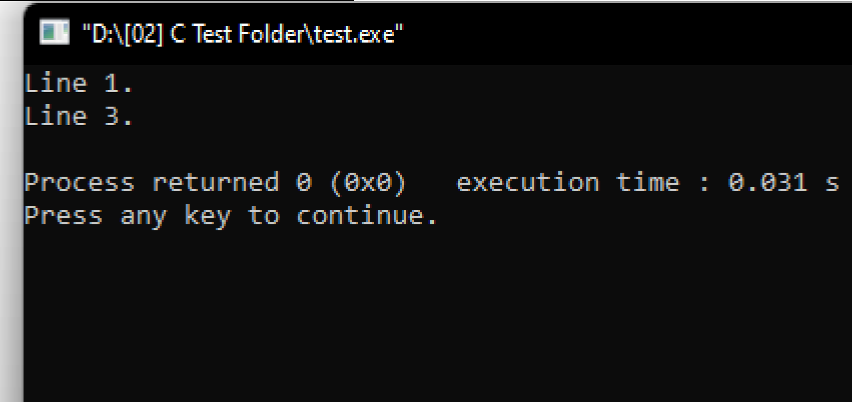
```c
#include <stdio.h>

int main()
{
    printf("Line 1. \n");
//    printf("Line 2. \n");
    printf("Line 3. \n");
}
```



```
Line 1.
Line 3.

Process returned 0 (0x0)   execution time : 0.031 s
Press any key to continue.
```

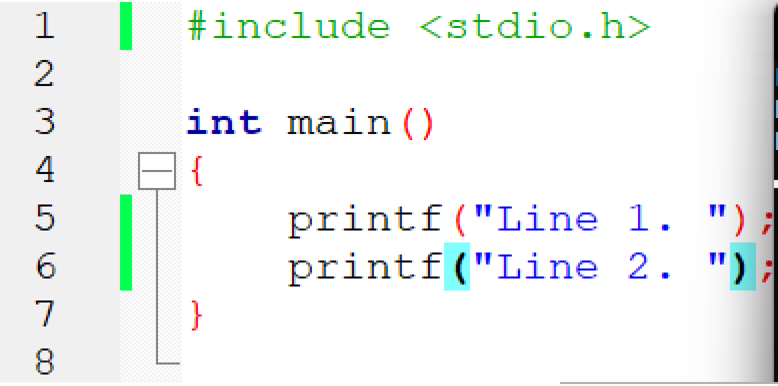


```c
#include <stdio.h>

int main()
{
    printf("Line 1. \n");
/*   printf("Line 2. \n"); */
    printf("Line 3. \n");
}
```

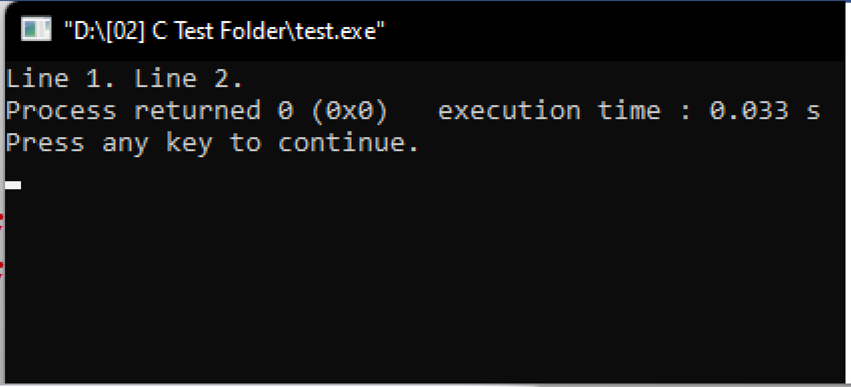


```
Line 1.
Line 3.

Process returned 0 (0x0)   execution time : 0.031 s
Press any key to continue.
```

PRINTF & PUTS



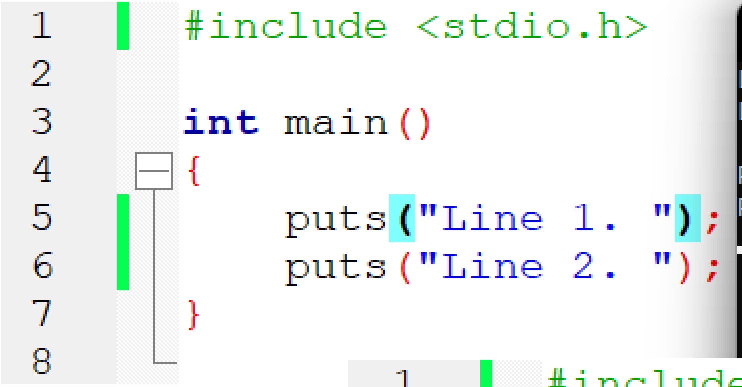
```c
#include <stdio.h>

int main()
{
    printf("Line 1. ");
    printf("Line 2. ");
}

```



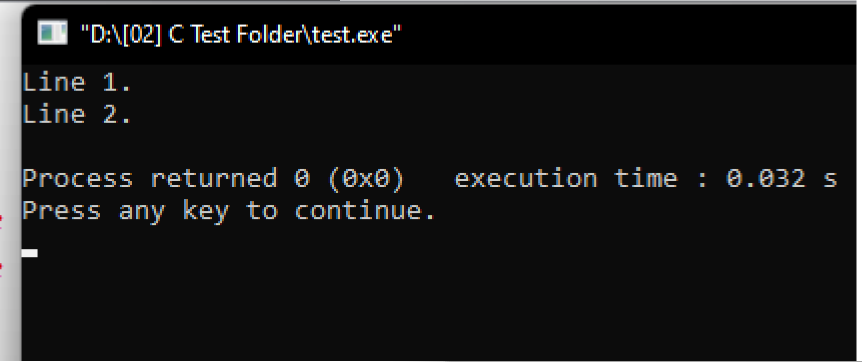
"D:\[02] C Test Folder\test.exe"

```
Line 1. Line 2.
Process returned 0 (0x0)   execution time : 0.033 s
Press any key to continue.
```

puts(“”); = printf(“\n”);

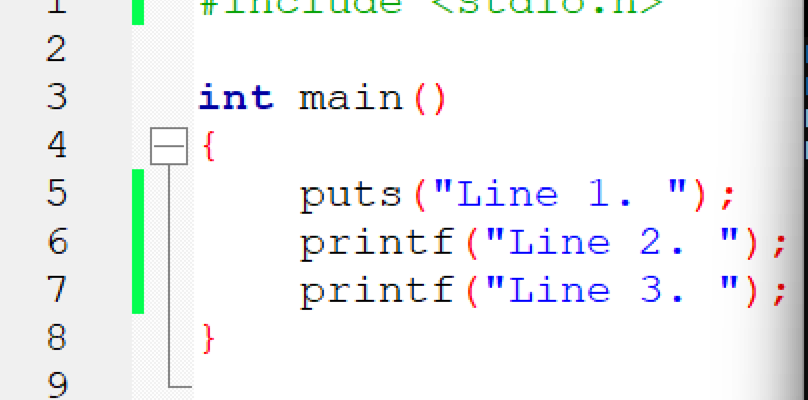


```c
#include <stdio.h>

int main()
{
    puts("Line 1. ");
    puts("Line 2. ");
}

```

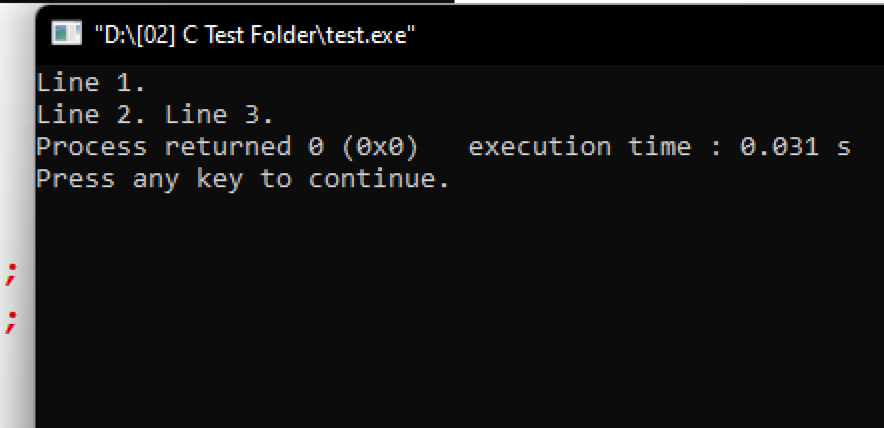


"D:\[02] C Test Folder\test.exe"

```
Line 1.
Line 2.

Process returned 0 (0x0)   execution time : 0.032 s
Press any key to continue.
```



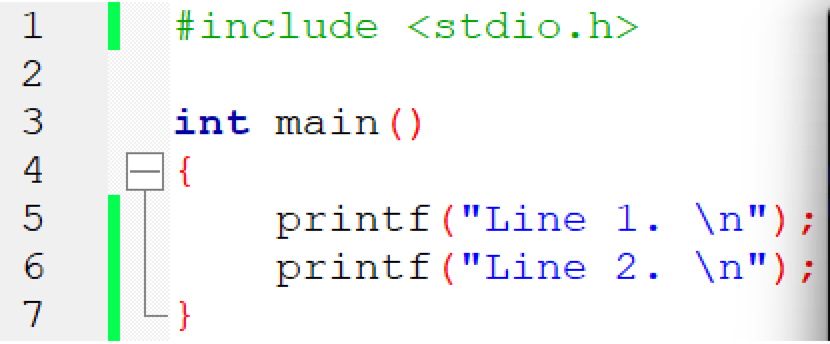
```c
#include <stdio.h>

int main()
{
    puts("Line 1. ");
    printf("Line 2. ");
    printf("Line 3. ");
}

```



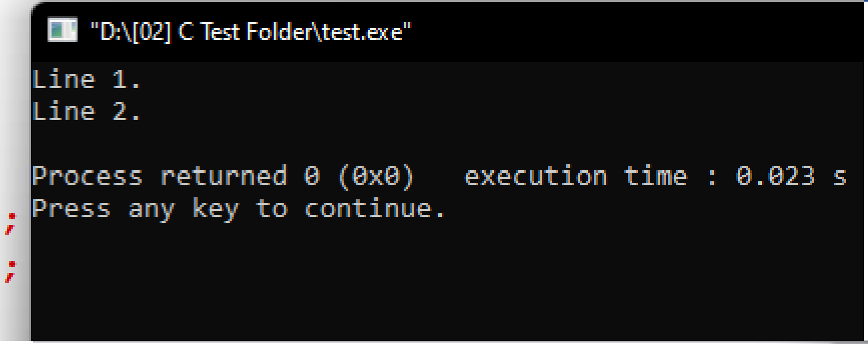
"D:\[02] C Test Folder\test.exe"

```
Line 1.
Line 2. Line 3.
Process returned 0 (0x0)   execution time : 0.031 s
Press any key to continue.
```

PRINTF & PUTS

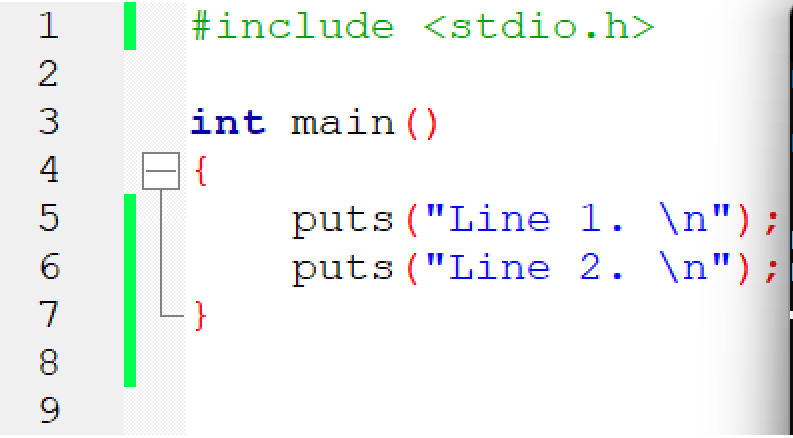


```c
#include <stdio.h>

int main()
{
    printf("Line 1. \n");
    printf("Line 2. \n");
}
```

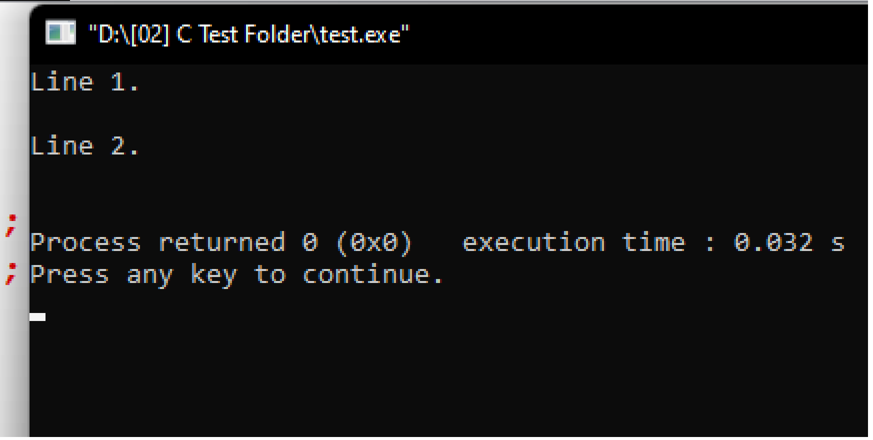


"D:\[02] C Test Folder\test.exe"

```
Line 1.
Line 2.

Process returned 0 (0x0)   execution time : 0.023 s
Press any key to continue.
```



```c
#include <stdio.h>

int main()
{
    puts("Line 1. \n");
    puts("Line 2. \n");
}


```

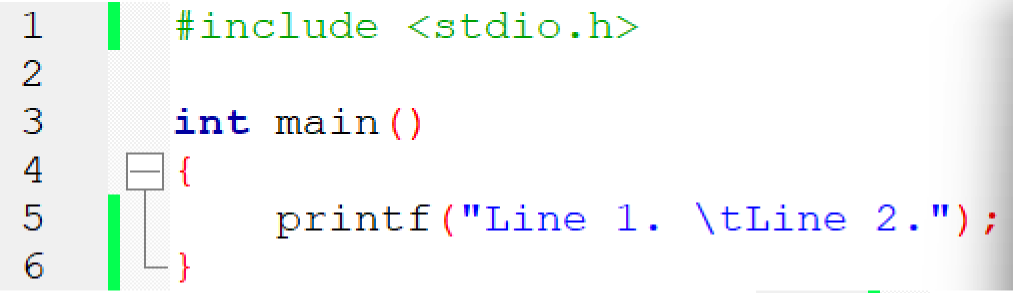


"D:\[02] C Test Folder\test.exe"

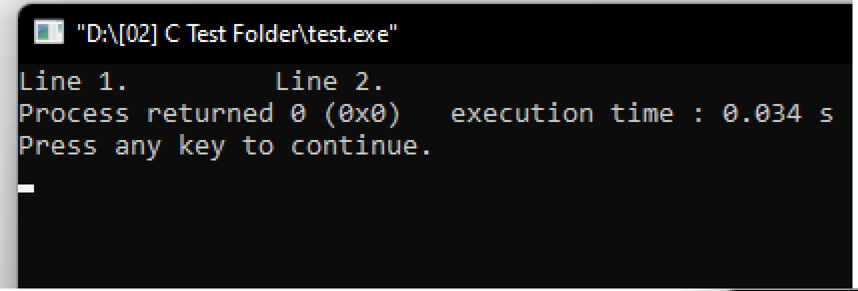
```
Line 1.

Line 2.


Process returned 0 (0x0)   execution time : 0.032 s
Press any key to continue.
```

\n ➔ new line

PRINTF & PUTS

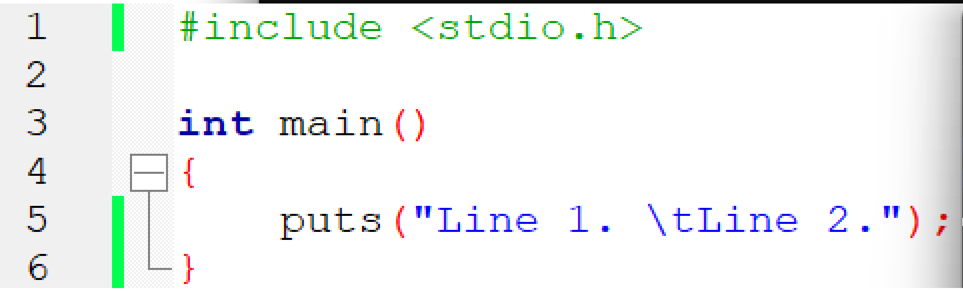


```c
#include <stdio.h>

int main()
{
    printf("Line 1. \tLine 2.");
}
```

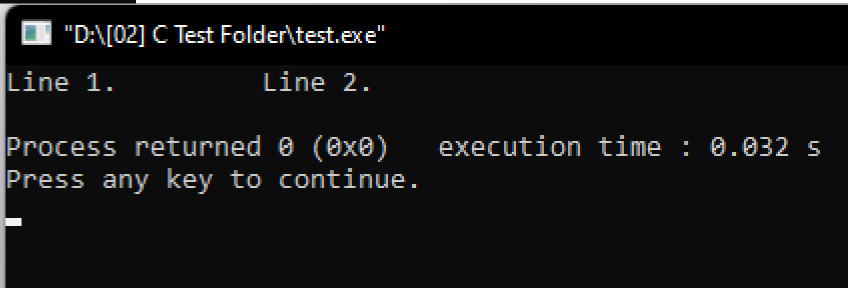


"D:\[02] C Test Folder\test.exe"

```
Line 1.         Line 2.
Process returned 0 (0x0)   execution time : 0.034 s
Press any key to continue.
```



```c
#include <stdio.h>

int main()
{
    puts("Line 1. \tLine 2.");
}
```

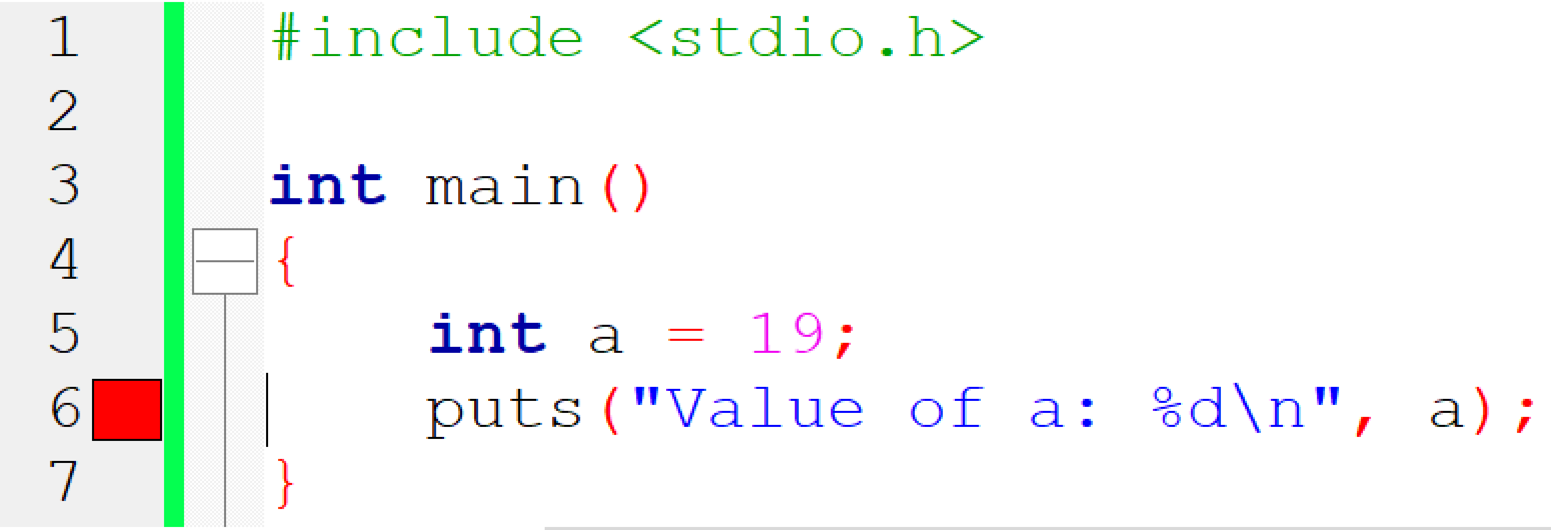


"D:\[02] C Test Folder\test.exe"

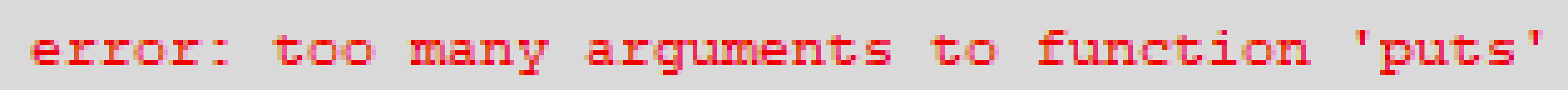
```
Line 1.         Line 2.

Process returned 0 (0x0)   execution time : 0.032 s
Press any key to continue.
```

\t ➔ tab

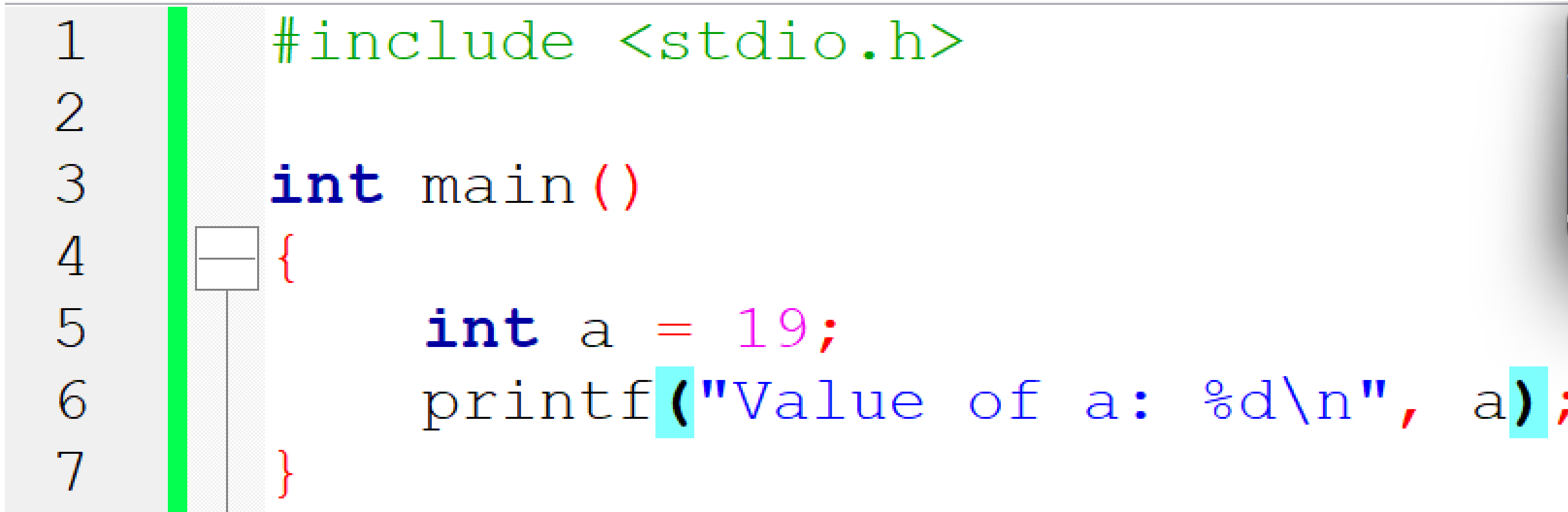
PRINTF & PUTS



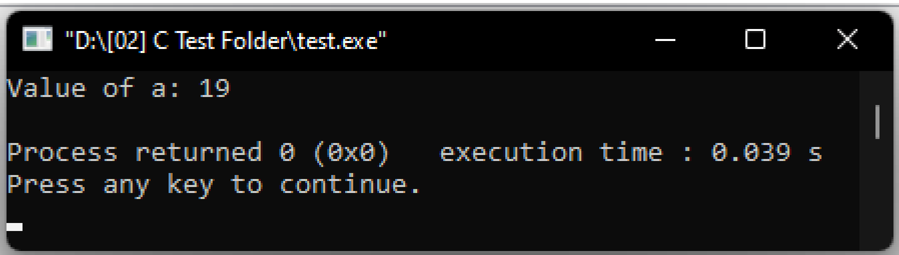
```c
#include <stdio.h>

int main()
{
    int a = 19;
    puts("Value of a: %d\n", a);
}
```



```
error: too many arguments to function 'puts'
```

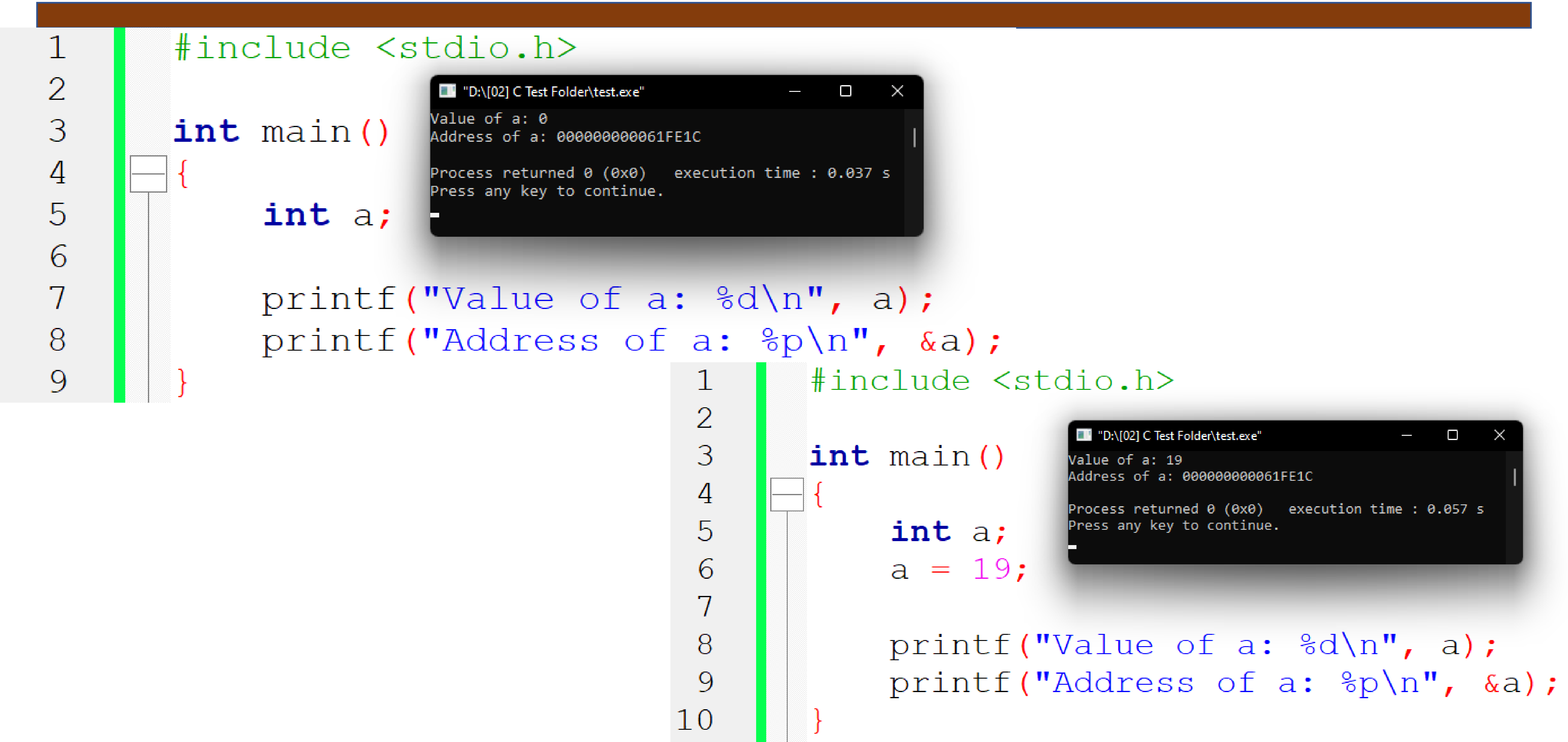


```c
#include <stdio.h>

int main()
{
    int a = 19;
    printf("Value of a: %d\n", a);
}
```



```
Value of a: 19

Process returned 0 (0x0)   execution time : 0.039 s
Press any key to continue.
```

DATA TYPES



```c
#include <stdio.h>

int main()
{
    int a;

    printf("Value of a: %d\n", a);
    printf("Address of a: %p\n", &a);
}
```

"D:\[02] C Test Folder\test.exe"

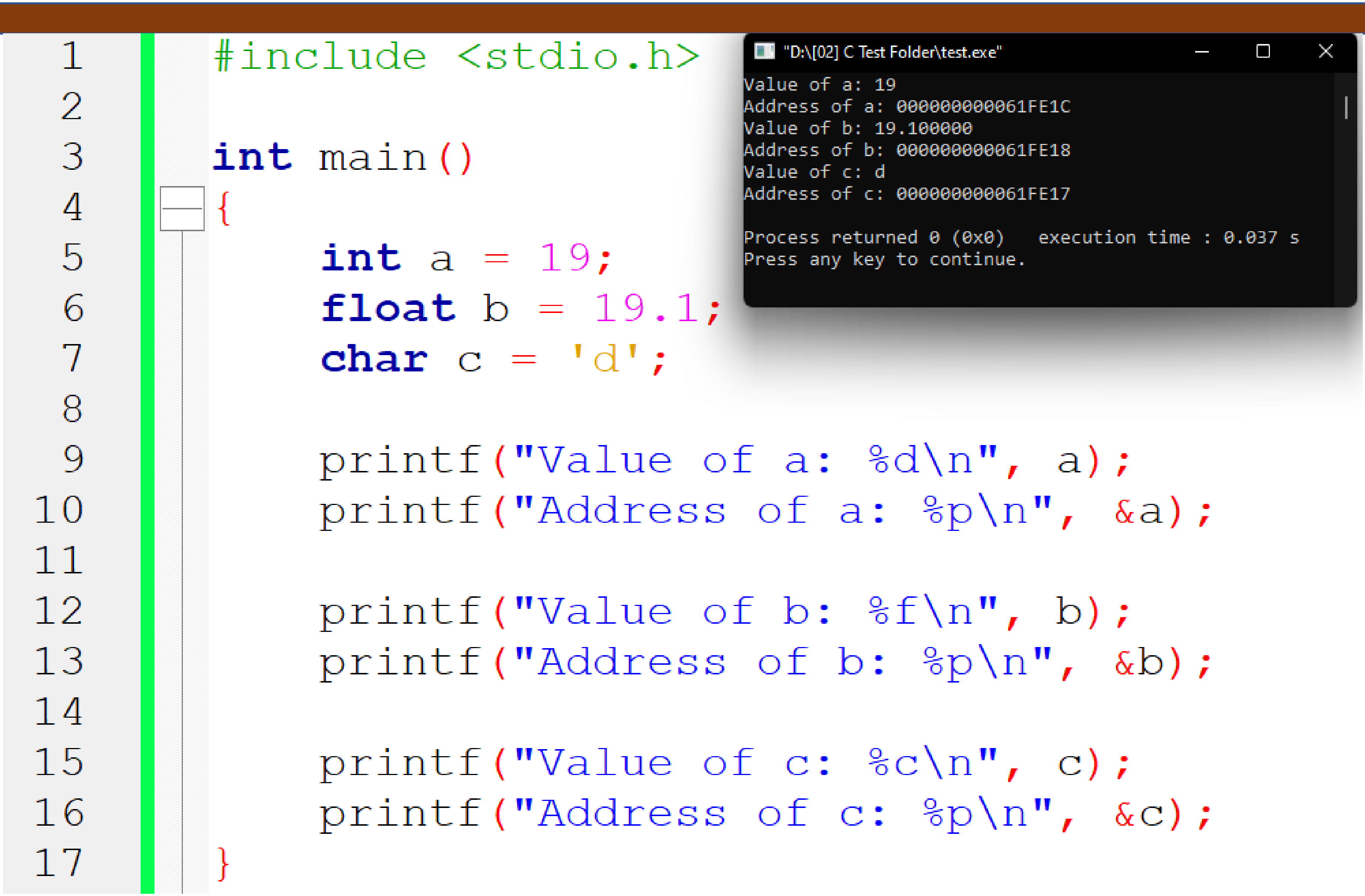
```
Value of a: 0
Address of a: 000000000061FE1C

Process returned 0 (0x0)   execution time : 0.037 s
Press any key to continue.
```

```c
#include <stdio.h>

int main()
{
    int a;
    a = 19;

    printf("Value of a: %d\n", a);
    printf("Address of a: %p\n", &a);
}
```

"D:\[02] C Test Folder\test.exe"

```
Value of a: 19
Address of a: 000000000061FE1C

Process returned 0 (0x0)   execution time : 0.057 s
Press any key to continue.
```

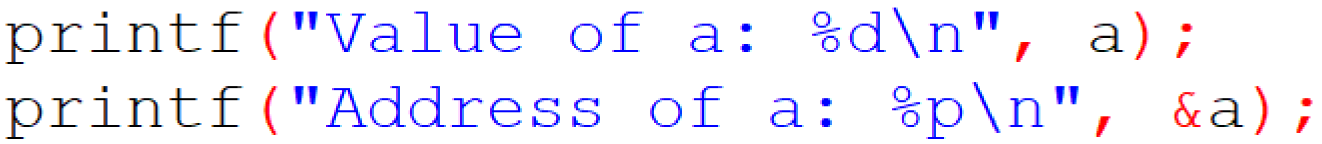
DATA TYPES



```c
#include <stdio.h>

int main()
{
    int a = 19;
    float b = 19.1;
    char c = 'd';

    printf("Value of a: %d\n", a);
    printf("Address of a: %p\n", &a);

    printf("Value of b: %f\n", b);
    printf("Address of b: %p\n", &b);

    printf("Value of c: %c\n", c);
    printf("Address of c: %p\n", &c);
}
```

```
Value of a: 19
Address of a: 000000000061FE1C
Value of b: 19.100000
Address of b: 000000000061FE18
Value of c: d
Address of c: 000000000061FE17

Process returned 0 (0x0)   execution time : 0.037 s
Press any key to continue.
```

DATA TYPES

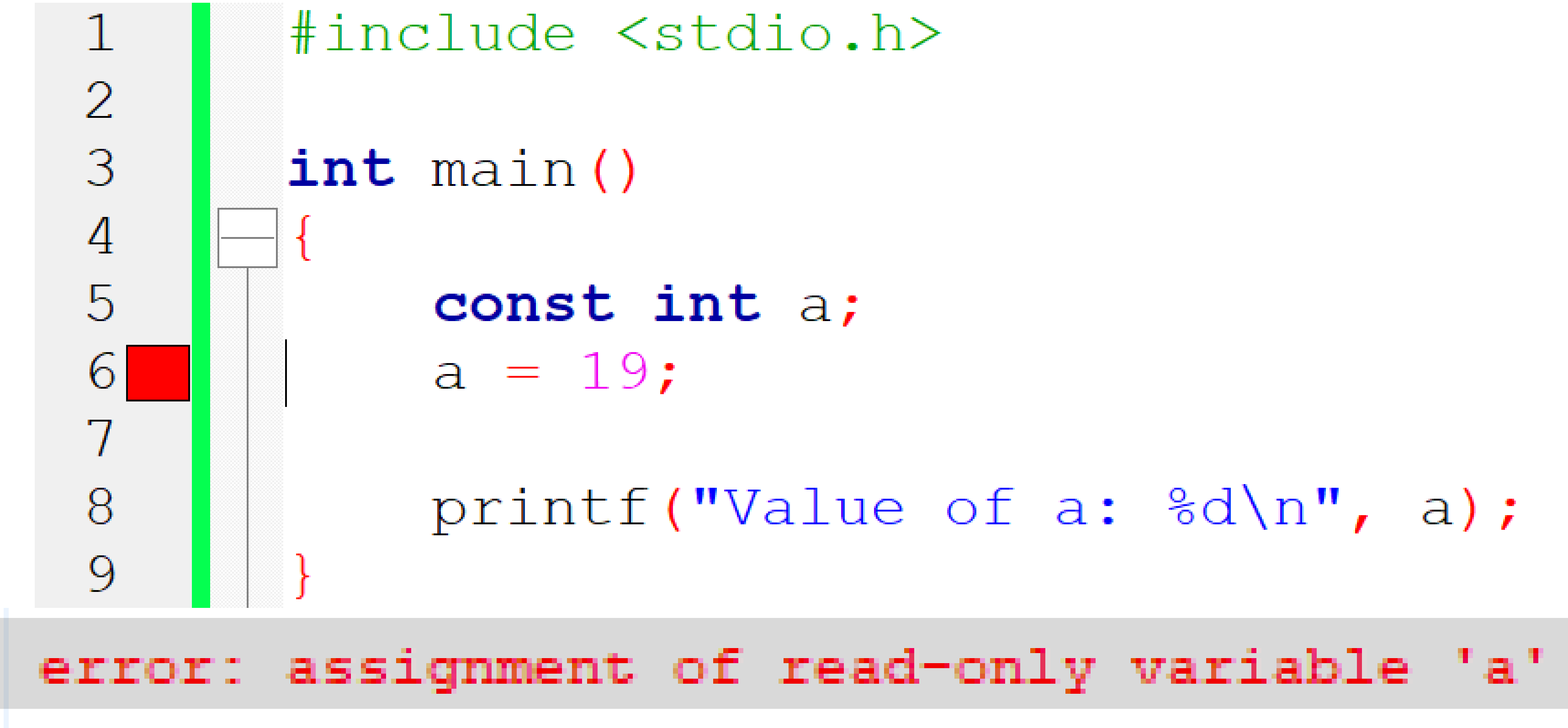


```
printf("Value of a: %d\n", a);
printf("Address of a: %p\n", &a);
```

%d → a

%p → &a

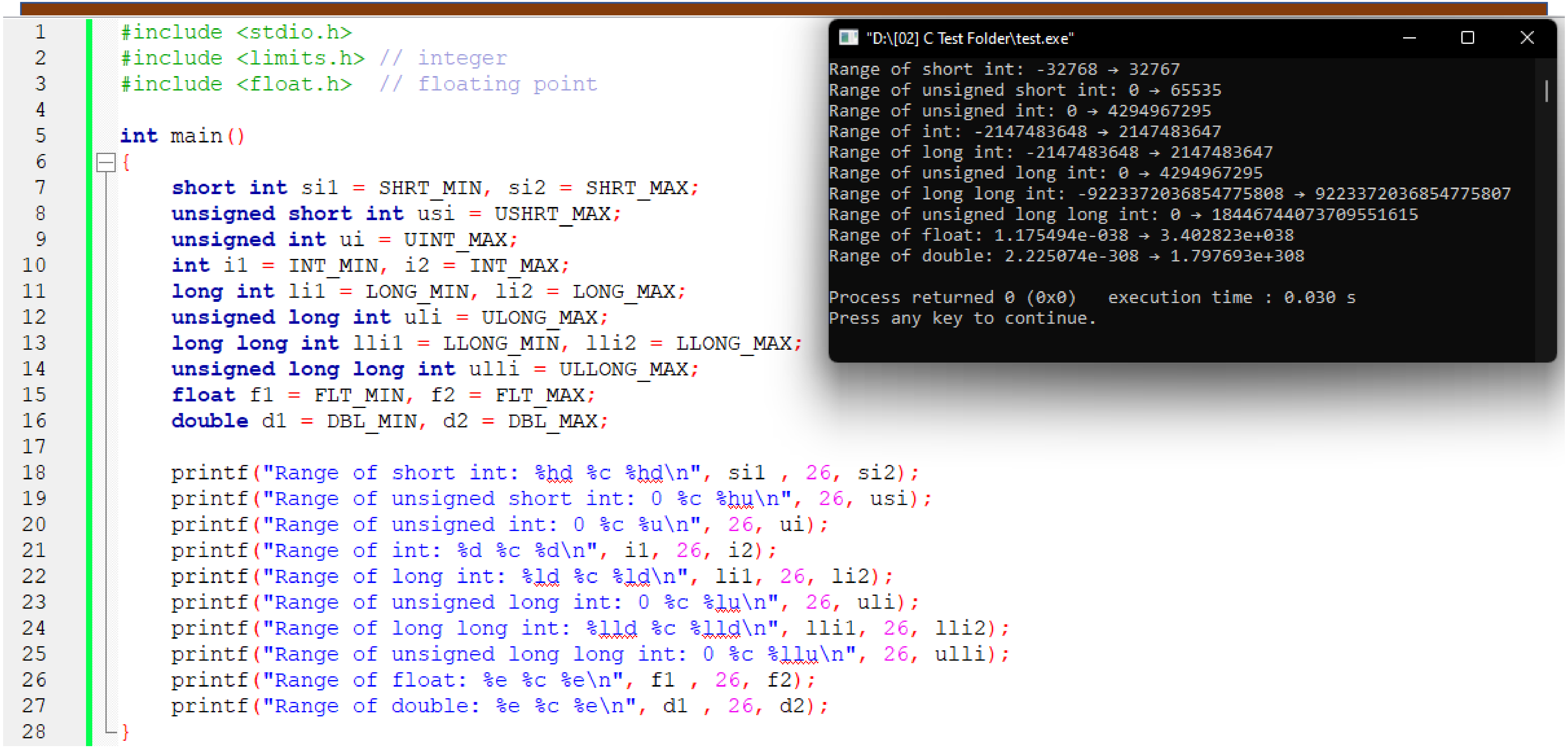
DATA TYPES



```c
#include <stdio.h>

int main()
{
    const int a;
    a = 19;

    printf("Value of a: %d\n", a);
}
```

```
error: assignment of read-only variable 'a'
```

DATA TYPES

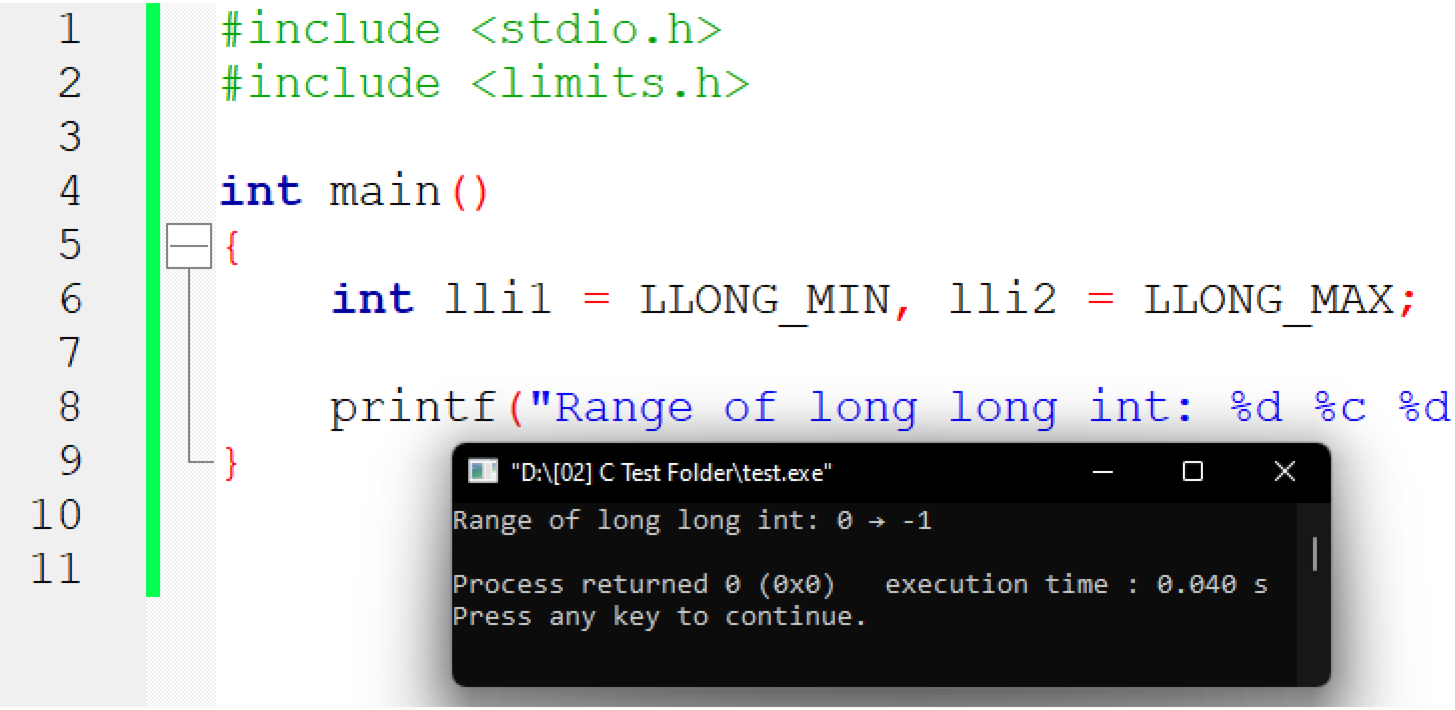


```c
#include <stdio.h>
#include <limits.h> // integer
#include <float.h>  // floating point

int main()
{
    short int si1 = SHRT_MIN, si2 = SHRT_MAX;
    unsigned short int usi = USHRT_MAX;
    unsigned int ui = UINT_MAX;
    int i1 = INT_MIN, i2 = INT_MAX;
    long int li1 = LONG_MIN, li2 = LONG_MAX;
    unsigned long int uli = ULONG_MAX;
    long long int lli1 = LLONG_MIN, lli2 = LLONG_MAX;
    unsigned long long int ulli = ULLONG_MAX;
    float f1 = FLT_MIN, f2 = FLT_MAX;
    double d1 = DBL_MIN, d2 = DBL_MAX;

    printf("Range of short int: %hd %c %hd\n", si1 , 26, si2);
    printf("Range of unsigned short int: 0 %c %hu\n", 26, usi);
    printf("Range of unsigned int: 0 %c %u\n", 26, ui);
    printf("Range of int: %d %c %d\n", i1, 26, i2);
    printf("Range of long int: %ld %c %ld\n", li1, 26, li2);
    printf("Range of unsigned long int: 0 %c %lu\n", 26, uli);
    printf("Range of long long int: %lld %c %lld\n", lli1, 26, lli2);
    printf("Range of unsigned long long int: 0 %c %llu\n", 26, ulli);
    printf("Range of float: %e %c %e\n", f1 , 26, f2);
    printf("Range of double: %e %c %e\n", d1 , 26, d2);
}
```

"D:\[02] C Test Folder\test.exe"

```
Range of short int: -32768 → 32767
Range of unsigned short int: 0 → 65535
Range of unsigned int: 0 → 4294967295
Range of int: -2147483648 → 2147483647
Range of long int: -2147483648 → 2147483647
Range of unsigned long int: 0 → 4294967295
Range of long long int: -9223372036854775808 → 9223372036854775807
Range of unsigned long long int: 0 → 18446744073709551615
Range of float: 1.175494e-038 → 3.402823e+038
Range of double: 2.225074e-308 → 1.797693e+308

Process returned 0 (0x0)   execution time : 0.030 s
Press any key to continue.
```

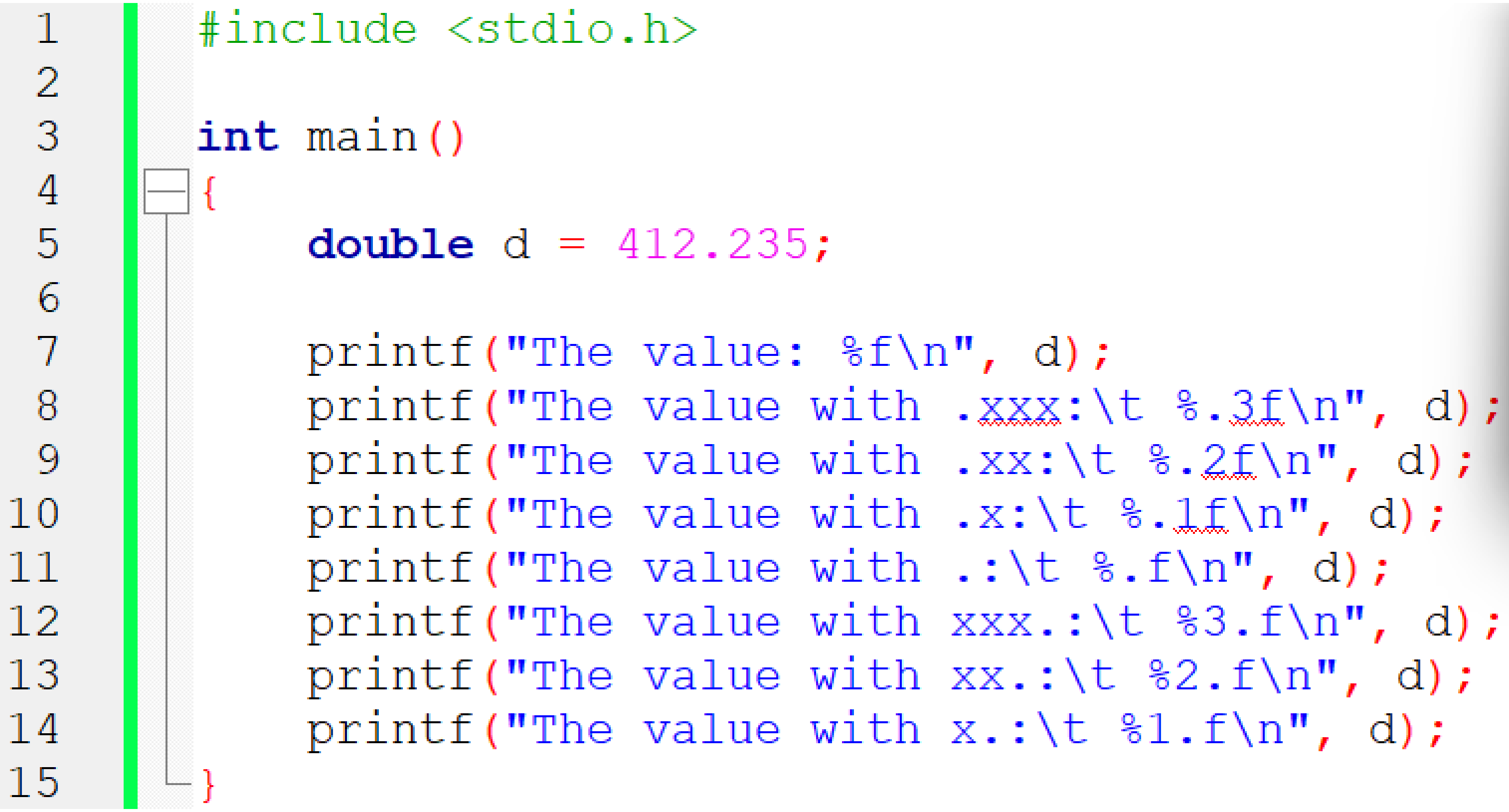
DATA TYPES



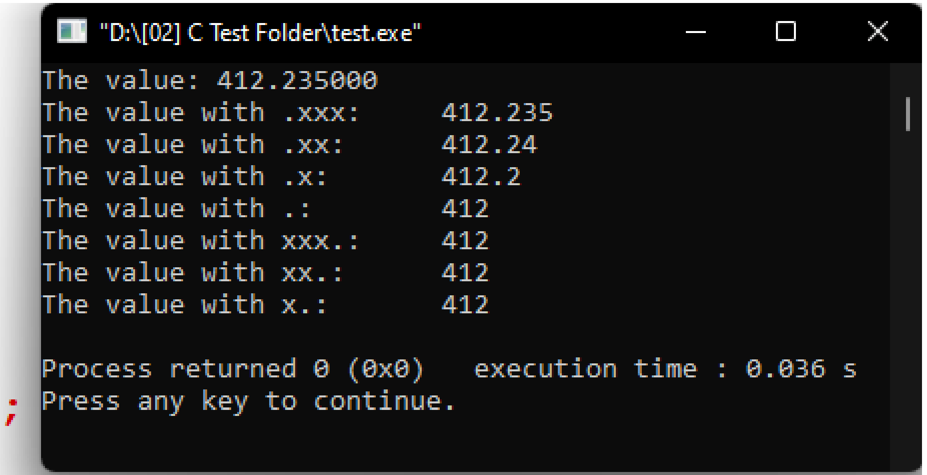
```c
#include <stdio.h>
#include <limits.h>

int main()
{
    int lli1 = LLONG_MIN, lli2 = LLONG_MAX;

    printf("Range of long long int: %d %c %d\n", lli1, 26, lli2);
}
```

```
Range of long long int: 0 → -1

Process returned 0 (0x0)   execution time : 0.040 s
Press any key to continue.
```

DATA TYPES

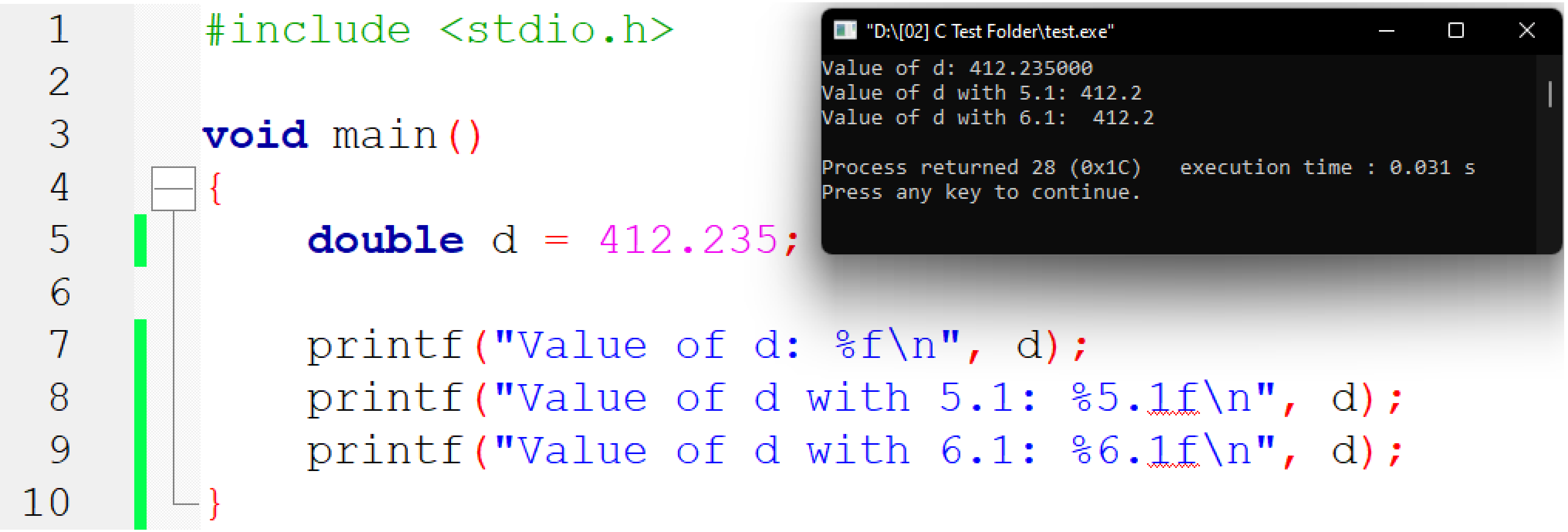


```c
#include <stdio.h>

int main()
{
    double d = 412.235;

    printf("The value: %f\n", d);
    printf("The value with .xxx:\t %.3f\n", d);
    printf("The value with .xx:\t %.2f\n", d);
    printf("The value with .x:\t %.1f\n", d);
    printf("The value with .:\t %.f\n", d);
    printf("The value with xxx.:\t %3.f\n", d);
    printf("The value with xx.:\t %2.f\n", d);
    printf("The value with x.:\t %1.f\n", d);
}
```



```
The value: 412.235000
The value with .xxx:     412.235
The value with .xx:      412.24
The value with .x:       412.2
The value with .:        412
The value with xxx.:     412
The value with xx.:      412
The value with x.:       412

Process returned 0 (0x0)   execution time : 0.036 s
Press any key to continue.
```

DATA TYPES

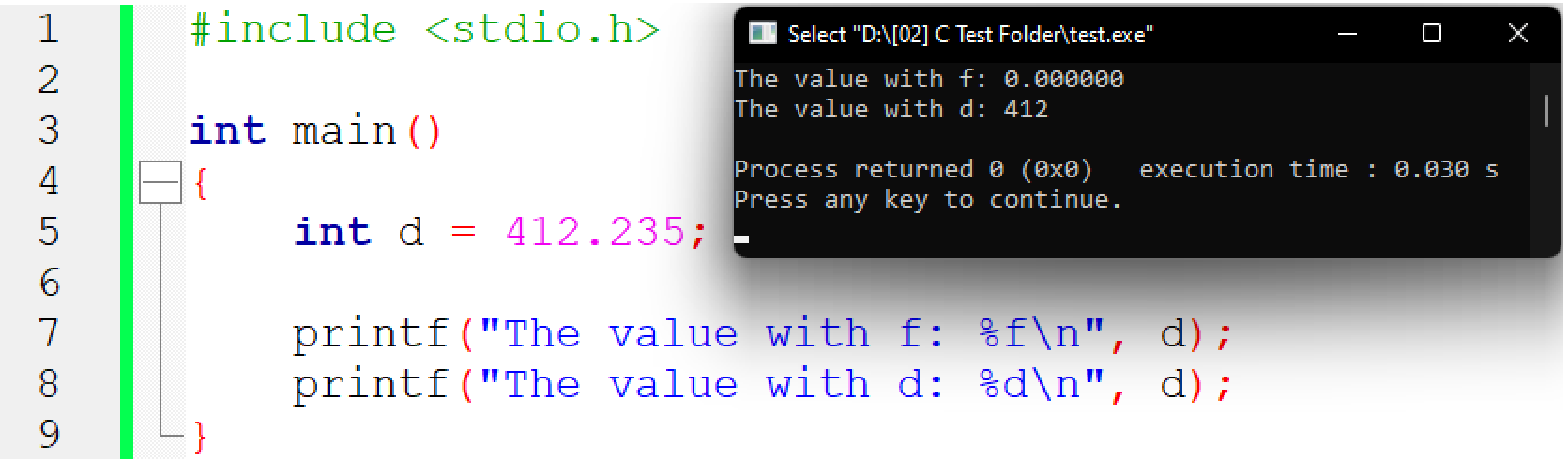


```c
#include <stdio.h>

void main()
{
    double d = 412.235;

    printf("Value of d: %f\n", d);
    printf("Value of d with 5.1: %5.1f\n", d);
    printf("Value of d with 6.1: %6.1f\n", d);
}
```

"D:\[02] C Test Folder\test.exe"

```
Value of d: 412.235000
Value of d with 5.1: 412.2
Value of d with 6.1:  412.2

Process returned 28 (0x1C)   execution time : 0.031 s
Press any key to continue.
```

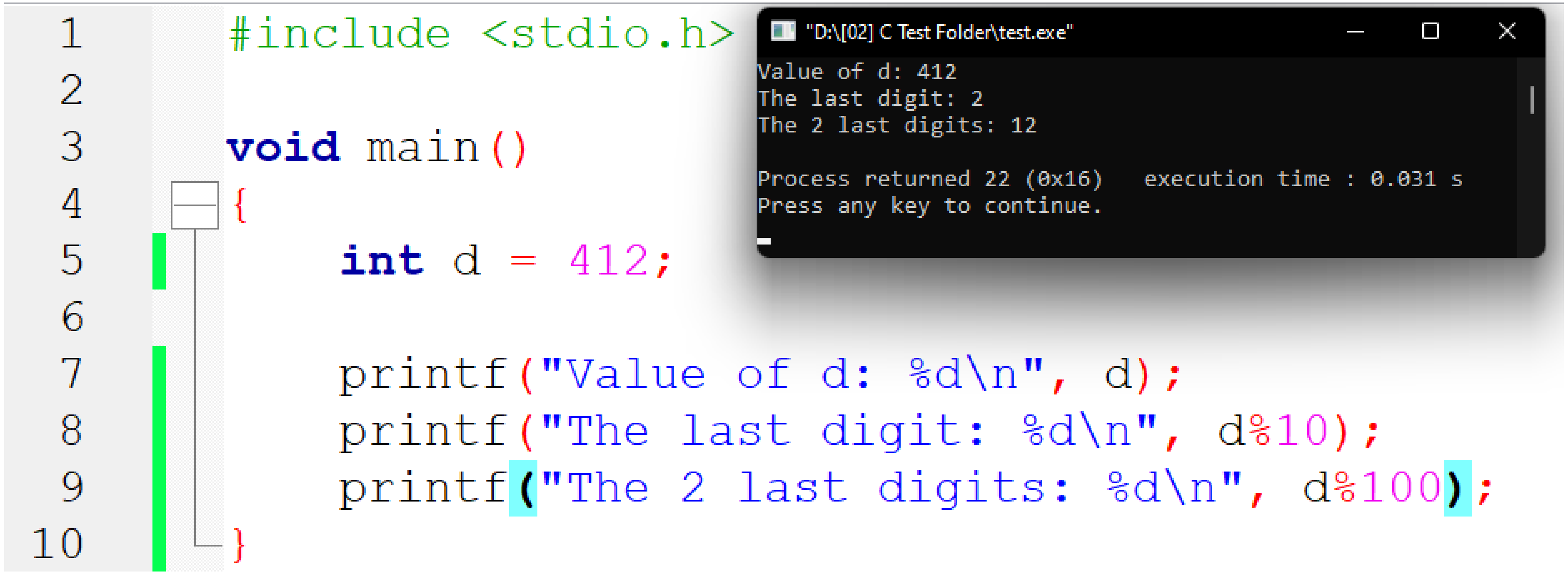
DATA TYPES



```c
#include <stdio.h>

int main()
{
    int d = 412.235;

    printf("The value with f: %f\n", d);
    printf("The value with d: %d\n", d);
}
```

```
The value with f: 0.000000
The value with d: 412

Process returned 0 (0x0)   execution time : 0.030 s
Press any key to continue.
```

DATA TYPES

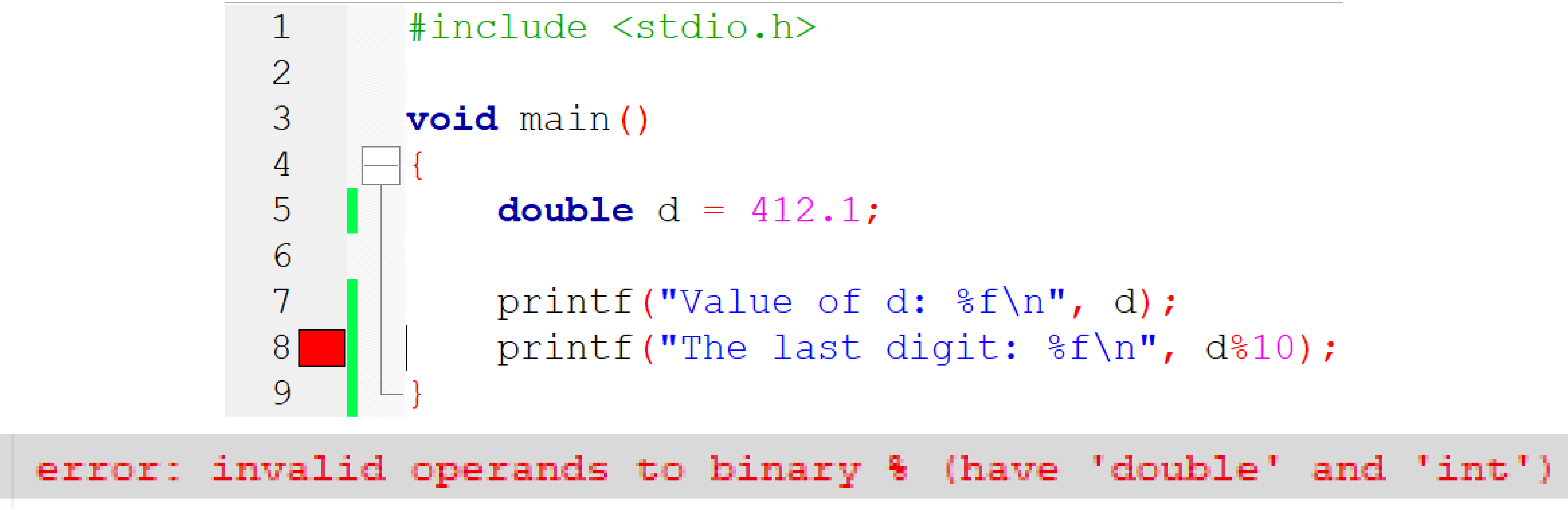


```c
#include <stdio.h>

void main()
{
    int d = 412;

    printf("Value of d: %d\n", d);
    printf("The last digit: %d\n", d%10);
    printf("The 2 last digits: %d\n", d%100);
}
```

"D:\[02] C Test Folder\test.exe"

```
Value of d: 412
The last digit: 2
The 2 last digits: 12

Process returned 22 (0x16)   execution time : 0.031 s
Press any key to continue.
```

DATA TYPES



```c
#include <stdio.h>

void main()
{
    double d = 412.1;

    printf("Value of d: %f\n", d);
    printf("The last digit: %f\n", d%10);
}
```

```
error: invalid operands to binary % (have 'double' and 'int')
```