

Tutorial 1

1. If you have a `float` variable set to 3.1, what would be the output of `printf`? Why?

In base 2, one tenth is an infinitely-repeating fraction (0.0001100110011...), so 3.1 cannot be represented exactly in binary.

2. If you have two `float` variables, say, `a` and `b`, how would you check if `a` equals `b` (floating-point equality)?

use something like

```
#include <math.h>

if(fabs(a - b) <= closeness * fabs(a))
```

The precise value of `closeness` have to be chosen carefully.

3. How would you round numbers (e.g., you are going to round a float variable: `x`)?

```
(int)(x < 0 ? x - 0.5 : x + 0.5)
```

4. Is the following code problematic? Or, what would be the value of `i`?

```
int i = 3;
i = i++;
```

The behaviour is undefined. If you want to increment `i`, use either `i=i+1`, `i+=1`, `i++`, or `++i`, not some combination.

5. Would the following code work?

```
int a = 10000, b = 10000;
long int c = a * b;
```

Use an explicit cast on at least one of the operands to force long arithmetic:

```
long int c = (long int)a * b;  
or  
long int c = (long int)a * (long int)b;
```

How about `(long int)(a * b)`?

6. See the code below. What is the value of `a`?

```
double a, b = 32.5;  
a = 2 / 5 * (b - 30);
```

0.

If you cast one of the operands to `float` or `double`, or use a floating-point constant, i.e.

```
a = (double)2 / 5 * (b - 30);  
or  
a = 2.0 / 5 * (b - 30);
```

it will work as you expect.

How about `(double)(2 / 5) * (b - 30)`?

7. Would the following code work?

```
long int n = 1000000;  
printf("%d\n", n);
```

Whenever you print long ints you must use the `l` modifier in the `printf` format (e.g. `%ld`). `printf` does not know the types of the arguments, so you must let it know by using the correct format specifiers.

8. Any problems with the following code?

```
int i;  
scanf("%d", i);  
or  
char s[10];  
scanf("%s", s);
```

In `scanf`, you always need a pointer. `i` is not a pointer, so you need `&i`. `s` is a pointer, so the second one is correct.

9. Would the following code work?

```
double d;  
scanf("%f", &d);
```

scanf uses %lf for double, and %f for float. Use %lf, or declare d as a float.

10. Would the following code work?

```
short int s;  
scanf("%d", &s);
```

With %d, scanf expects a pointer to an int. To convert to a short int, use %hd.