

L1: Intro to Variables, Data Types

Monday, January 26, 2026 10:24 AM

Variables and Data Types

- Variables are a storage for information, and are specific to a certain data type.

1 byte: 8 bits;

example byte of information: 10010011

These bytes are then interpreted by the computer.

Integers are 4 bytes, and therefore 32-bits.

3 properties: 1. Name - reference

2. Type - what kind of data

3. Value - what information stored.

All values stored in memory somewhere.

Syntax is (:

Type name = value;

e.g. int age;

int age = 20;

8 bytes;
'double' of int

→ double gpa = 3.5;

1 byte

→ char grade = 'A';

%d int

%f floating point

%i signed decimal integer

%p Address (hex)

Type	Size	Range
------	------	-------

1 - p 1001001 (near)

Type	table	Type	Size	Range
		Char	1 byte	-128 to 127
		int	4 bytes	± 2 billion
		long	8 bytes	± 9 quintillion

int count = 100;
 int temp = -10;
 long worldPopulation = 800000000L;

The size of a variable dictates how many numbers can be represented.

Floats w/ 4 bytes have a precision of ~7 digits.

Type	Specifier	Example
int	%d	printf("%d", 42);
long	%ld	printf("%ld", 1234);
float/double	%f	printf("%f", 3.14f);
char	%c	printf("%c", 'A');

```
1 #include <stdio.h>
2
3 int main(){
4     int age = 20; //ints are 4 bytes
5     double gpa = 3.75; // double is 8 bytes, float is 4 bytes
6     char grade = 'Z'; // char is 1 byte
7
8     printf("Age: %d\n", age);
9     printf("GPA: %.2f\n", gpa);
10    printf("Grade: %c\n", grade);
11
12    printf("sizeof(char) = %zu\n", sizeof(char));
13    printf("sizeof(int) = %zu\n", sizeof(int));
14    printf("sizeof(double) = %zu\n", |
15    // comment
16    /** ..... */
17    return 0;
18 }
```

sizeof operator