

Doubles/floats

- Range of +/- 1.7e308
- 16 digits of precision
- Overflow = +/- inf, underflow = 0
- Double = 8b, float = 4b
- Floats are less precise, used in Arduino

Integers

- Max value is +/-2147483647
- 2 or 4 bytes

Vari changes

- Short int x: half the byte size of int (half range)
- Long int x: double byte size of int (double range)
- Unsigned: positive numbers (double range)

Functions

- Usually double, int, void (depends on what you want to return)
- type _name(what it inputs){}

Arduino

- void setup () is used to initialize variables and configurations
- pinMode (pin1, OUTPUT) means that the pin will have an output role
- Serial.begin(115200) initializes for print out
- void loop () runs the code repeatedly, like while(1)
- use float instead of double
- digitalWrite(pin1, HIGH) sets the pin to have 5V
- Serial.print() will print to the IDE
- micros()*1.0e-6 is time in s
- digital is 0V or 5V, analog is continuous

Arrays/dynamic

- Always starts at 0
- Dynamic array is only for 1D
- 2D arrays: A[row][column]
- Size must be an int
- A[] is allowed, not A[][]

Arrays/dynamic (cont)

- A[i]++; means increase the value of A[i] by 1
- Dynamic array:
double *A;
int n; // which we'll use for input
A = new double [n];
if(A == NULL){
// check if it exists and can run code
cout << "nerror in dynamic allocation";
exit(1);
}
delete[] A;
A = nullptr;

Char/strings

- Used for strings
- String is char array
- Max range of - 128/127
- '\0' is the term char
- char a = "a"
- For space, ' '
- Always make char size 100 or 1000

- Reverse string:
void string_reverse(char str1[], char str2[]){
int i=0;
int counter=0;

while(str1[i] != '\0'){
counter++;
i++; // this finds string length
}

for(i=0;i<counter;i++){
str2[i] = str1[counter -1 -i]; // because at counter its '\0'
}
str2[counter] = '\0'; // to stop the string
}

Files

- ofstream fout // writing to file
- ifstream fin // reading file
- fout.close() at end of file part
- create file: fout.open(" .txt")
- have #include <fstream>

- Check error:
if (!fout or !fin){
cout << "nerror with file";
exit(1);

Special info

- Break to stop loop
- Bool has 1 = true and 0 = false
- Exit(0) to stop program
- && means and
- Dot product: sum+= A[i] * B[i]
- || means or
- cout <<scientific << setprecision(9)
- sizeof(int) =4
- sizeof(double) = 8
- sizeof (bool) = 1
- sizeof(float)=4

- Call by reference:
&x means that it brings changes to main, not a copy of x

Examples

- Transpose:
for (i=0;i <3; i++){
for (j=0;j <3; j++){
tA[i][j] = A[j][i];
}
- Dynamic array:
double *p1;
n = 10 or cin >> n;
p1 = new double [n];
if (p1 == NULL) return 1;
delete p1;
- No_space:
void no_space (char str1 []) {
int i, newsize=0;
for (i=0; str1[i]; i++){
if (str1[i]!=' '){
str 1[n ew s -
ize]=s tr1[i];
new size++;
}
}
str 1[n ew s ize]='\0';
}
- Altern ating string:
j = 0;
for(i= 0;i <n _min ;i++) {
str3[j] = str1[i]; j++;
str3[j] = str2[i]; j++;
}
for(i= n _m in; i<n _ma x;i++) {
if(n_max == n1) str3[j] =
str1[i];

```
}
```

Pointers

- Can store anything (int, double, function, etc)
- `int*b` checks what's inside b
- `p = &b` means p is b's address
- `*(&b)=b` is simply the address of b
- `*p=11` means there's 11 stored inside b
- `*(p+1)` increments the pointer
- `*p + 1` increments what's inside p
- For char, it prints everything after the address
- For double it prints what's at the specific address



By **codinghippo**

cheatography.com/codinghippo/

Not published yet.

Last updated 4th December, 2025.

Page 1 of 3.

Sponsored by **CrosswordCheats.com**

Learn to solve cryptic crosswords!

<http://crosswordcheats.com>

Examples (cont)

```
> else str3[j] = str2[i]; j++;
```

```
}
```

```
str3[j] = '\0';
```

Arduino example:

```
float t0, t, v = 30;
int p;
int input0;
float v0;
// a)
servo1.attach(3);
servo1.write(0);
delay(1000); // wait for the servo
// b) (30 deg / s)
t0 = micros()*1.0e-6; // initial time
// p = p0 + v(delta_t) = p0 + v(t-t0)
// trajectory
while(1) {
  t = micros()*1.0e-6;
  p = v*(t-t0); // trajectory
  servo1.write(p);
  if(p >= 90) break;
}
while(1) {
  input0 = analogRead(A0);
  v0 = (input0/1023.0)*5;
  // convert int to volts
  if(v0 > 3) break;
}
servo1.write(0);
delay(1000);
exit(0);
}
```



By **codinghippo**

cheatography.com/codinghippo/

Not published yet.

Last updated 4th December, 2025.

Page 2 of 3.

Sponsored by **CrosswordCheats.com**

Learn to solve cryptic crosswords!

<http://crosswordcheats.com>