C++ Basic Cheat sheet for competitive programming

Prince Billy Graham Karmoker
@princebillyGK

Format specifiers

```
%[flags][min field width][precision][length]conversion specifier
  -----  --------------   ---------  ------ ------------------
   \             #,*        .#, .*     /             \
    \                                 /               \
   #,0,-,+, ,',I                 hh,h,l,ll,j,z,L   c,d,u,x,X,e,f,g,s,p,%
   -------------                 ---------------   ----------------------
   # | Alternate,                 hh | char,           c | unsigned char,
   0 | zero pad,                   h | short,          d | signed int,
   - | left align,                 l | long,           u | unsigned int,
   + | explicit + - sign,         ll | long long,      x | unsigned hex int,
     | space for + sign,           j | [u]intmax_t,    X | unsigned HEX int,
   ' | locale thousands grouping,  z | size_t,         e | [-]d.ddde±dd double,
   I | Use locale's alt digits     t | ptrdiff_t,      E | [-]d.dddE±dd double,
                                   L | long double,  --------=====
   if no precision   => 6 decimal places            /  f | [-]d.ddd double,
   if precision = 0  => 0 decimal places      _____/   g | e|f as appropriate,
   if precision = #  => # decimal places               G | E|F as appropriate,
   if flag = #       => always show decimal point      s | string,
                                             ..............------
                                            /          p | pointer,
   if precision      => max field width    /           % | %
```

Datatype and their range

```
    Char → char        8 bits      -128 to 127 or 0 to 255
           char_16_t   16 bits
           char_32_t   32 bits
           wchat_t     represent any largest supported characterset

    int  -> int            -2147483648 to +2147483647 or 0  to  4294967295
         -> short int      -32768 to +32767 or  0  to  65535
         -> long int       | -9223372036854775808 to +9223372036854775807
         or long long int  | or   0  to  18446744073709551615


    float -> float          ±1.17549e-38 to ±3.40282e+38
          -> double         ±2.22507e-308 to ±1.79769e+308
          -> long double    ±3.4E-4932 to ±1.1E+4932

    bool  -> bool
    void  -> void
    Null pointer  -> decltype(nullptr)
```

Initialization

```
int foo = 100;
int goo(200);
int hoo{300};
//automatic datatype automatically detects the data types
auto ioo = 100;
//null pointer initialization automatically detects the data type
decltype(ioo) joo;
int foo[10] = {1,2,3} //other elements will be automatically init to 0
int foo[10] {1, 2,3} //method 2 with out = sign
```

Octal and Hexadecimal Literals

```
0x44    hexadecimal
044     octal
```

Function Arguments

```
one dimentional typename func(typename var[]);
multi dimentional typename func(typename var[][x][y]);
return_type (*function_name)(argument_type arg1,…);
```

Pointer

```
*p++ //increment pointer
*++p //increment pointer
++*p //increment the value its points to
(*p)++ //increment the value its points to
```

const pointer

```
const int * p; //pointer is non-const but the value its point to can not be
changed using the pointer
int const * p; // pointer is const and the value its point to can be changed
using the pointer
```

pointer to pointer

```
int a = 44;
int *b = &a;
int **c = &b;
int ***d = &c;
int ****e = &d;
int *****f = &e;
```

here:

```
f == &e
*f == e == &d
**f == *e == d == &c
***f == **e == *d == c == &b
****f == ***e == **d == *c == b == &a
*****f == ****e  ***d == **c == *b == a == 44
```