**Competitive programming key notes**

**Type casting:**

**s2.append(to\_string(cnt))** is used to append a character to string. **To\_string(int)** is used to convert integer into string. **Stoi(s)** is used to convert string to integer.

stoi("3.14159") is 3

stoi("31337 geek") is 31337

similarly **atoi()** takes char array or string as input.

This is the code to convert encode int to char: char((A - 1) % 26 + 'A')

Or char(65) = ‘A’

**Ascii code:**

A – 65

a – 97

0 – 48

**Accessing structure members:**

Struct node{

Int var1;

Int var2;

}

In main

Struct node temp, ptr\*;

Temp.var1 == (\*ptr).var1 == ptr->var1;

**Referencing in fuctions:**

Void fun(int &ref) ------this takes the addresss of actual val

Void fun(int ref) -------this takes the copy of actual val.

We call both functions as fun(ref).

Call by pointer:

Void fun(int\* ref) ----

Call this as fun(x) where int\* x = &ref;

**Max value assignment :**

Way 1 :

#include<limits>

long long int INF = numeric\_limits<long long int>::max();

way 2:

LLONG\_MAX

**Initialization of a vector:**

vector<vector< pair<long long int, long long int>>> al;

al.assign(n, vector< pair<long long int, long long int>>());

**Sorting a vector of pairs:**

Use the following function in sort() method

bool sortbysec(const pair<int,float> &a,

              const pair<int,float> &b)

{

    return (a.second < b.second);

}

And use sort()

sort(m.begin(), m.end(), sortbysec);

this will sort in asending order. To sort in decending order use

(a.second > b.second);

In **sortbysec** function.

**Subtracting 1 to a number:**

All the bits after rightmost set bit flips.

10 in binary is 00001010  
9 in binary is 00001001  
8 in binary is 00001000  
7 in binary is 00000111