

CP Snippets

[About](#)[Codeforces](#)[GitHub](#)[LinkedIn](#)

About

A collection of CPP Snippets to aid in competitive programming.

This site was auto generated with the help of [marked](#).

```
curl -L "https://raw.githubusercontent.com/theSoberSobber/CP-Sr
```

Index -

- [DSU](#) : DSU
- [arr-inp](#) : arr-inp
- [arr-pref](#) : arr-pref
- [binpow](#) : binpow
- [binsearch](#) : binsearch
- [bp](#) : bp
- [clock_for_TL](#) : clock
- [combination-non-mod](#) : combination-non-mod
- [combination-small](#) : combination-small
- [combination](#) : combination
- [crt](#) : crt
- [derangments](#) : derangments
- [diophantine](#) : linear diophantine
- [dsu-rr](#) : dsu-rr
- [easy_seive](#) : easy_seive
- [euclid](#) : euclid

- **explanation_binsearch** : explanation_binsearch
- **fac** : fac
- **factorization** : factorization
- **fenwick** : binary indexed tree
- **file_io** : for coding competitions
- **freq-map** : freq-map
- **gr-inp-Fwt** : graph input weight
- **gr-inp** : graph input
- **highest_exponent** : power_in_fac
- **interactive** : essential measures for interactive problems
- **ip-overloads** : I/O Overloads that I don't use
- **kadane** : max subarray sum $O(n)$
- **kosaraju** : kosaraju
- **kruskal** : kruskal
- **lambda_function** : lambda_function
- **lca** : LCA path satisfying some condition
- **log** : log
- **matrix** : matrix
- **mint** : modular integer
- **modpow** : modpow
- **pbds** : pbds
- **pq** : pq
- **recur-binsearch** : recursive binary search implementation to make intuition easier ig
- **recur-modpow** : recur-modpow
- **rng** : rng
- **segtree** : sextree
- **seive** : seive
- **tokenizer** : tokenizer that has no use
- **totient-seive** : totient-seive
- **totient** : totient
- **trie** : trie

- **troll** : troll
- **two-sat (kosaraju)** : two-sat (kosaraju)
- **xor-basis** : xor-basis