**Programação Competitiva**

Passos : [página ref](http://lampiao.ic.unicamp.br/maratona/?name=preparacao)1.~~Escolher a linguagem.~~

* + Dominar sintaxe e comandos básicos.
  + Dominar as funções mais usadas em competições.

1. Noções de Complexidade de Algoritmos
   * Calcular complexidade de tempo e memória.
   * Estimativa de tempo.
2. Dominar as estruturas de dados básicas.
   * Vetores, strings.
   * Pilha, fila.
3. Dominar Entrada e Saída
   * Leitura dos diferentes tipos.
   * Saída formatada dos diferentes tipos.
4. Ficar à vontade com Recursão e Backtracking.
   * Saber enumerar permutações, combinações e arranjos.
5. Aprender Grafos.
   * Estruturas para representá-los.
   * Algoritmos de busca: BFS, DFS.
   * Árvore Geradora Mínima.
   * Caminhos mínimos.
6. Aprender Programação Dinâmica.
   * Entender a idéia.
   * Conhecer os problemas clássico.

Anotações Gerais :

#include <bits/stdc++.h>

using namespace std;

int main(){

return 0;

}

tam vetor = sizeof(vetor)/sizeof(vetor[0])

swap(x,y) => troca x por y e y por x

sort(v, v+p) => ordena o vetor apontado por v da posição 0 até p

min(x,y) => retorna o menor valor entre x e y  
max(x,y) => retorna o maior valor entre x e y

reverse(v,v+p) => imprime ao contrário

template<typename T>

struct par{

T x,y;

}

se declararmos   
par<int> p; Substituimos T por int

par<double> q; Substituimos T por double

vector<int> v; Vetor de inteiros v vazio, melhor opção

v.push\_back(k); adiciona o elemento k no final do vetor

string s = “abacaba”

s += “é um palindromo”;

string s = “abacaba é um palíndromo”;

Stack/Pilha :

stack <tipodado> nome\_variavel;

stack <string> cartas;

cartas.size(); TAMANHO STACK

cartas.push(“Rei de Copas”); ADICIONA ELEMENTO  
cartas.push(“A de espadas”);

cartas.pop(); RETIRA O ELEMENTO DO TOPO, no caso o “A de Espadas”

cartas.top(); IMPRIME A CARTA NO TOPO

cartas.empty(); RETORNA TRUE SE VAZIA E FALSE SE TEM PELO MENOS 1 ELEMENTO

Fila/Queue :

queue <tipodado> nome\_variavel;

queue <string> fila;

fila.size(); tamanho da fila

fila.push(“Thiago”); adiciona o thiago na fila  
fila.push(“Beltrano”); adiciona beltrano na fila

fila.pop() ;remove a pessoa na frente da fila (thiago)

fila.front(); imprime (não remove) a primeira carta

fila.back(); imprime o ultimo elemento d afila