Moscow Institute of Physics and Technology

My Pity

Fedor Alekseev, Dmitry Ivaschenko, Daria Kolodzey

Whatever contest

today

Strings (1)

AhoCorasick.h

Description: on-line tracking of the set of suffixes of a text that are prefixes of some words from a dictionary.

44 lines

```
struct AhoCorasick {
  AhoCorasick(): n(1) {
    n.reserve(TrieSize);
  }

  void addWord(const string& word, int id) {
    int v = 0;
    for (int ch : word) {
      ch -= 'a';
      auto& u = n[v].trans[ch];
      if (!u) {
        u = int(n.size());
        n.emplace_back();
      }
      v = u;
    }
    n[v].termId = id;
  }

  void build() {
    queue<int> q;
    for (q.push(0); !q.empty(); q.pop()) {
      auto v = q.front();
      for (Char ch = 0; ch < Alph; ++ch) {
        auto& u = n[v].trans[ch];
        if (!u) {
          u = n[n[v].link].trans[ch];
          continue;
        }
        q.push(u);
        auto i = n[u].link = (v ? n[n[v].link].trans[ch] : 0);
        n[u].nextTerm = (n[i].termId >= 0 ? i : n[i].nextTerm);
      }
    }
  }

private:
  struct Node {
    int trans[Alph]{};
    int nextTerm = -1, termId = -1, link = 0;
  };

  vector<Node> n;
};
```

PrefixFunction.h

Description: pi[x] is the length of the longest prefix of s that ends at x, other than s[0..x] itself

10 lines

```
vector<size_t> pi(const string& s) {
  vector<size_t> p(s.size(), 0);
  for (size_t i = 1; i < s.size(); ++i) {
    auto px = p[i - 1];
    while (px && s[i] != s[px])
      px = p[px - 1];
    p[i] = px + (s[i] == s[g]);
  }
  return p;
}
```

ZFunction.h

Description: z[x] is max L: s[x:x+L] == s[:L]

11 lines

```
vector<size_t> zFun(const string& s) {
  vector<size_t> z(s.size(), 0);
  for (size_t left = 0, right = 0, i = 1; i < s.size(); ++i) {
    z[i] = (i < right ? min(right - i, z[i - left]) : 0);
    while (i + z[i] < s.size() && s[i + z[i]] == s[z[i]])
      ++z[i];
    if (i + z[i] > right)
      tie(left, right) = {i, i + z[i]};
  }
  return z;
}
```