Solução OPEI 2017 - Modalidade C

Problema A

```cpp
#include <bits/stdc++.h>

using namespace std;

int main() {
  int N, a;
  cin >> N;
  if (!N) {
    cout << "100.00%";
    return 0;
  }
  int sum = 0;
  vector<int> vec;
  for(int i = 0; i < N; i++) {
    cin >> a;
    vec.push_back(a);
    sum += vec[i];
  }
  double med = (double) sum / (double) N;
  int cnt = 0;
  for(int i =0; i < N; i++) {
    if ((double) vec[i] >= med) ++cnt;
  }
  double ans = (double) cnt / (double) N;
  printf("%.2lf%%", 100.0 * ans);
  return 0;
}
```

Problema B

```cpp
#include <bits/stdc++.h>

using namespace std;

int main() {
  int N;
  cin >> N;
  double H = ((N * sqrt(6.0)) / 3.0);
  double V = ((N * N * N) * sqrt(2.0) / 12.0);
  double A = N * N * sqrt(3.0);
  printf("%.2lf %.2lf %.2lf\n", H, A, V);
  return 0;
}
```

Problema C

```cpp
#include <bits/stdc++.h>

using namespace std;

int N;

bool is_inc_sorted(const vector<int> &v) {
  bool ans = true;
  for(int i = 1; i <= N; i++)
    ans = ans && v[i-1] == i;
  return ans;
}

bool is_dec_sorted(const vector<int> &v) {
  bool ans = true;
  for(int i = N, k = 0; i >= 1; i--, k++)
    ans = ans && v[k] == i;
  return ans;
}

bool is_full(const vector<int> &v) {
  map<int, int> has;
  for(int i = 0; i < N; i++) has[v[i]] = 1;
  bool ans = true;
  for(int i = 1; i <= N; i++)
    ans = ans && has[i];
  return ans;
}

int main() {
  cin >> N;
  vector<int> vec;
  int a;
  for(int i=0 ; i < N; i++) {
    scanf("%d", &a);
    vec.push_back(a);
  }
  if (is_inc_sorted(vec)) cout << "PERMUTACAO CRESCENTE";
  else if (is_dec_sorted(vec)) cout << "PERMUTACAO DECRESCENTE";
   else if (is_full(vec)) cout << "PERMUTACAO COMUM";
  else cout << "ERRO";

  return 0;
}
```

Problema D

```cpp
#include <bits/stdc++.h>

using namespace std;

int main() {
  int N;
  string s;
  cin >> N;
  getchar();
  getline(cin, s);
  N %= 26;
  for(int i =0 ; i < s.size(); i++) {
    if (s[i] == ' ') continue;
    if (s[i] >= 'a' && s[i] <= 'z') {
      if (N + s[i] > 'z') s[i] = 'a' + N + s[i] - 'z';
      else s[i] += N;
    } else {
      if (N + s[i] > 'Z') s[i] = 'A' + N + s[i] - 'Z';
      else s[i] += N;
    }
  }
  cout << s;
}
```

Problema E

```cpp
#include <bits/stdc++.h>
#define fr(a, b, c) for(int a = b; a < c; ++a)

using namespace std;

int main() {
  int N, M, K;
  cin >> N >> M >> K;
  int mat[500][500];
  memset(mat, 0, sizeof mat);

  int a, b, c, d;
  fr(i, 0, K) {
    cin >> a >> b >> c >> d;
    a--, b--, c--, d--;
    fr(j, a, c+1) fr(k, b, d+1) {
      mat[j][k] = 1;
    }
  }
  int cnt = 0, tot = N * M;
  fr(i, 0, N) fr(j, 0, M) if (mat[i][j]) cnt++;

  double ans = (double) cnt / (double) tot;
  printf("%.2lf%%", 100.0 * ans);
  return 0;

  return 0;
}
```

Problema F

```cpp
#include <bits/stdc++.h>
#define fr(a, b, c) for(int a = b; a < c; ++a)

using namespace std;

typedef long long ll;

int main() {
  string str, a, b;
  cin >> str;
  fr(i, 0, str.size()) {
    if (str[i] == '?') {
      a.push_back('0');
      b.push_back('9');
    } else {
      a.push_back(str[i]);
      b.push_back(str[i]);
    }
  }

  int j = 0;
  while (j < a.size() && a[j] == '0') a[j] = '1';

  ll v1, v2;
  v1 = stoll(a), v2 = stoll(b);

  cout << llabs(v1 - v2) << endl;

  return 0;
}
```

Problema G

```c
#include <bits/stdc++.h>

using namespace std;

int main(){
    int n, q;
    scanf("%d %d", &n, &q);
    vector<int> stars(n, 0);
    for(int i = 0; i < n; i++){
        scanf("%d", &stars[i]);
    }
    for(int i = 0; i < q; i++){
        int k;
        scanf("%d", &k);
        if(k == 1){
            int x;
            scanf("%d", &x);
            stars[x-1]++;
        } else {
            int l, r;
            scanf("%d %d", &l, &r);
            int k = 0;
            for(int a = l-1; a <= r-1; a++){
                k += stars[a];
            }
            printf("%d\n", k);
        }
    }
    return 0;
}
```

Problema H

```c
#include <stdio.h>
#include <cstring>

int main(){
    char in[1000001];
    scanf("%s", in);
    int l = 0;
    int r = strlen(in)-1;
    bool palindrome = true;
    while(l < r){
        if(in[l] != in[r]){
            palindrome = false;
        }
        l++; r--;
    }
    if(palindrome){
        printf("S\n");
    } else {
        printf("N\n");
    }
    return 0;
}
```

Problema I

```cpp
#include <bits/stdc++.h>

using namespace std;

int main() {
  int N;
  cin >> N;
  int O, P, Q;
  vector<int> v;
  int index, maxi = 0;
  for(int i = 1; i <= N; i++) {
    cin >> O >> P >> Q;
    if (Q) P <<= 1;
    v.push_back(P);
    maxi = max(maxi, P);
  }
  for(int i = 1; i <= N; i++)
    if (v[i - 1] == maxi)
      index = i;
  vector<int> ans;
  for(int i = 1; i <= N; i++)
    if (i != index)
      ans.push_back(i);
  ans.push_back(index);
  for(int i = 0; i < N; i++) {
    if (i) cout << ' ';
    cout << ans[i];
  }
  return 0;
}
```

Problema J

```cpp
#include <bits/stdc++.h>

using namespace std;

int main() {
    while(1)
    {
        int h1, m1, h2, m2;
        cin >> h1;
        cin >> m1;
        cin >> h2;
        cin >> m2;
        if(!h1 && !m1 && !h2 && !m2)
            break;
        int m = (m2 - m1 + 60) % 60;
        int h = (h2 - h1 + 24) % 24;
        if (m1 > m2)
            h = (h2 - h1 + 24 - 1) % 24;

        h *= 60;
        cout << m + h << endl;
    }


    return 0;
}
```