

競技プログラミング向けの プログラミング言語 Albat

鳥海翔一

- ・Albat という自作言語をやっています。

(<https://sechack365.nict.go.jp/achievement/2024/pdf/24Dk.pdf>)

[illegible]

やってること

SecHack365^[1]というハッカソンで開発していた

競技プログラミング向けのプログラミング言語 Albat^[2] という言語の続き

・動機

競プロにおいては、C++が標準とされているけど、

自作のライブラリやマクロが長すぎたり 、includeファイルや名前空間の記述

など競プロするなら面倒な要素があったりする

※1 <https://sechack365.nict.go.jp/>

※2 <https://sechack365.nict.go.jp/achievement/2024/pdf/24Dk.pdf>

デモ的なやつ 1

コード例

例題

整数 N, K と長さ N の整数列 P, Q が与えられます。

P と Q から1つずつの整数を選び、その合計を K にすることは できますか？

答えを出力するプログラムを作成してください。

説明

入出力

@変数1, 変数2 ... → 入力 としてFastIOが呼び出されます。vectorとかstringの入力も行えます。
\$(hoge) → 出力を行います。\$\$にすることで改行出力になります。

ライブラリ

yesno(arg1, str1, str2) → arg1のTrue, False に応じてstr1かstr2が返されます。デフォルトではyesかnoを返

その他 (展開)

vi : vector<int> , fore(x, X) : for(auto &x : X)

```
1 #include <iostream>
2 using namespace std;
3 int N, K;
4 int P[109], Q[109];
5 bool ok = false;
6 int main() {
7     cin >> N >> K;
8     for (int i = 1; i <= N; i++) cin >> P[i];
9     for (int i = 1; i <= N; i++) cin >> Q[i];
10    for (int x = 1; x <= N; x++) {
11        for (int y = 1; y <= N; y++) {
12            if (P[x] + Q[y] == K) ok = true;
13        }
14    }
15    if (ok == true) cout << "Yes" << endl;
16    else cout << "No" << endl;
17    return 0; }
```

C++

17行

```
1 N, K = map(int, input().split())
2 P = list(map(int, input().split()))
3 Q = list(map(int, input().split()))
4 ok = False
5
6 for x in range(N):
7     for y in range(N):
8         if P[x] + Q[y] == K:
9             ok = True
10
11 if ok == True:
12     print("Yes")
13 else:
14     print("No")
```

Python

14行

```
1 int@N, K;
2 vi@(P(N), Q(N));
3 int ok = 0;
4 fore(p, P) fore(q, Q) ok |= (p + q == K);
5 $$$(yesno(ok));
```

Albat

5行!!!

↑これが

C++に翻訳されると...

こうなる→

```
me@PC124-003:~/a$ tail -n 30 a.cpp
#define rd io.read

string yesno(int ok, string yes = "Yes", string no = "No"){
    return ok ? yes : no;
}

int main(){
    int N, K;
    rd(N);
    rd(K);
    vector<int> P(N), Q(N);
    rd(P);
    rd(Q);
    int ok = 0;
    for(auto &p : P){
        for(auto &q : Q){
            ok |= (p + q == K);
        }
    }
    wtl<n(yesno(ok));
    return 0;
}

//---endLn---

// --- original ---
// int@N, K;
// vi@(P(N), Q(N));
// int ok = 0;
// fore(p, P) fore(q, Q) ok |= (p + q == K);
// $$$(yesno(ok));
```

デモ的なやつ 2

例題

前スライドの例題が hoge 件与えられるから全部解いてください

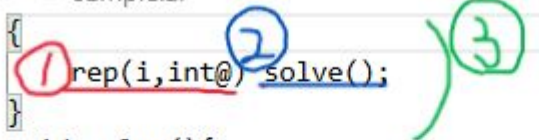
デモ的なやつ 2

例題

前スライドの例題が hoge 件与えられるから全部解いてください

testcase > ≡ sample.al

```
1  {
2  rep(i, int@) solve();
3  }
4  void solve(){
5      int@N,K;
6      vi@(P(N), Q(N));
7      int ok = 0;
8      fore(p, P) fore(q,Q) ok |= (p + q == K);
9      $$yesno(ok);
10 }
```



1. 適当な変数で入力を受け取れる
→ hoge回だけsolve()を呼び出せばいい
2. 後方定義の関数を呼び出せる
→ プロトタイプ宣言不要！
3. main関数の補完をしてくれる

他

・競技プログラミングするときに使うような機能

→ 演算子 + α , 条件分岐, 繰り返し, 関数定義, ラムダ式 とか

Albat独自の演算子

```
1 //演算子の例
2 int a = 4, b = 6;
3
4 //べき乗
5 $$a ** b; // 4096
6
7 //2点間のユークリッド距離を求める
8 int x1 = 1, y1 = 2, x2 = 3, y2 = 5;
9 $$sqrt((x1 - x2) ** 2 + (y1 - y2) ** 2); //3.6055..
```

```
11 //余り切り上げの割り算
12 $$a /+ b; // 1
```

```
14 //負の数を正の数で割った余りを非負で返す
15 $$((a - b) % b); // -2
16 $$((a - b) %+ b); // 4
17
18 //リングバッファとしてarrを扱うとき
19 vector<int> arr = {1, 2, 3, 4, 5, 6};
20 $$arr[(a-b)%b]; //a[-2]は未定義動作
21 $$arr[(a-b)%+b]; //a[4] = 5
```

競技プログラミングで使いそうなデータ構造

```
1 //UnionFind
2 UnionFind uf(100);
3
4 //SegTree
5 int op(int a, int b) { return max(a, b); }
6 int e() { return -1; }
7 vector<int> A(100, 0);
8 _SegTree<int, op, e> seg(A);
9
10 //modint
11 modint a = 1;
12
13 //Mexを高速で求めるライブラリ
14 MexSet s;
15 etc ...|
```

競技プログラミングでよく見るようなマクロ(展開されます)

```
INFI // 2**30 - 2**15 = 1073709056
INFL // 2**62 - 2**31 = 4611686016279904256

MOD // デフォルトでは 1000000009

ll // long long
vi // vector<int>
P // pair<int, int>
```