**Готуємось до олімпіади з інформатики 2019-1**

|  |  |  |
| --- | --- | --- |
| № | Методи побудови алгоритмів | Приклади |
|  | Числові ряди | НСД \_\_gcd(a,b)НСК a\*b/\_\_gcd(a,b)ФібоначіКаталана<https://www.e-olymp.com/uk/problems/230>Прості числа<https://www.e-olymp.com/uk/problems/2479> |
|  | Алгоритми переведення чисел з однієї позиційної системи числення в іншу | <https://www.e-olymp.com/uk/problems/3952><https://www.e-olymp.com/uk/problems/4835> |
|  | Структури даних: стек, черга, список, дерево, множини | <https://www.e-olymp.com/uk/problems/555><https://www.e-olymp.com/uk/problems/2661><https://www.e-olymp.com/uk/problems/3004><https://www.e-olymp.com/uk/problems/1872><https://www.e-olymp.com/uk/problems/2040><https://www.e-olymp.com/uk/problems/1228><https://www.e-olymp.com/uk/problems/1226> |
|  | Пошукові алгоритми: лінійний, бінарний | <https://www.e-olymp.com/uk/problems/4117><https://www.e-olymp.com/uk/problems/4669><https://www.e-olymp.com/uk/problems/3451><https://www.e-olymp.com/uk/problems/5908>  |
|  | Методи сортування | #include<algorithm>sort(a+0,a+n); |
|  | Алгоритми роботи з довгими числами | Дії в стовпчикЕволюція<https://www.e-olymp.com/uk/problems/223> |
|  | Комбінаторика | ПерестановикиРозміщенняКомбінації#include <iostream>#include <algorithm>using namespace std;int main(){int n;cin>>n;int a[n+1];for(int i=1;i<=n;i++)a[i]=i;for(int i=1;i<=n;i++)cout<<a[i]<<" ";cout<<"\n"; while (next\_permutation(a+1,a+n+1)){for(int i=1;i<=n;i++)cout<<a[i]<<" ";cout<<"\n";} return 0;}<https://www.e-olymp.com/uk/problems/65><https://www.e-olymp.com/uk/problems/390><https://www.e-olymp.com/uk/problems/1787> |
|  | Теорія графів | <http://www.e-olymp.com/uk/problems/4764> - Матриця суміжності, степінь вершин<http://www.e-olymp.com/uk/problems/4763> - Від списку ребер до матриці суміжності<http://www.e-olymp.com/uk/problems/625> - Пошук в глибину на графах<http://www.e-olymp.com/uk/problems/975> - Флойд (зчитування матриці)<http://www.e-olymp.com/uk/problems/983> - Флойд (створення матриці)<http://www.e-olymp.com/uk/problems/2968> - Флойд (Форд)<http://www.e-olymp.com/uk/problems/1365> - Дейкстри<http://www.e-olymp.com/uk/problems/2965> - Дейкстра<http://www.e-olymp.com/uk/problems/981> - мінмальне остове дерево (алгоритм Прима)<http://www.e-olymp.com/uk/problems/964> - Матриця інцендентності |
|  | Динамічне програмування | «Задача про найбільшу спільну підпослідовність» «Задача про рюкзак»<https://www.e-olymp.com/ru/problems/799>флойд<https://www.e-olymp.com/uk/contests/8160>for(k=1;k<=n;k++)for(i=1;i<=n;i++)for(j=1;j<=n;j++)a[i][j]=min(a[i][j].a[i][k]+a[k][j]; |
|  | Жадібні алгоритми | <https://www.e-olymp.com/uk/problems/80><https://www.e-olymp.com/uk/problems/138><https://www.e-olymp.com/uk/problems/182>  |
|  | Обчислювальної геометрії | Підрахувати кількість лівих поворотів.int x[100],y[100],a[100],b[100],v[100];int i,n;//зчитування координатcin>>n;for(int i=0;i<n;i++) cin>>x[i]>>y[i];//обчислення координат векторівfor(int i=0;i<n-1;i++) {a[i]=x[i+1]-x[i];b[i]=y[i+1]-y[i];}//обчислення векторних добутківfor(int i=0;i<n-2;i++) v[i]=a[i]\*b[i+1]-a[i+1]\*b[i]; //підрахунок кількості лівих поворотів векторівint s=0;for(int i=0;i<n-2;i++) if(v[i]>0) s++;cout<<s++<<endl;<https://www.e-olymp.com/uk/problems/1685><https://www.e-olymp.com/uk/problems/2148><https://www.e-olymp.com/uk/problems/4279>  |