	GUC German University in Caird الجامعة الألمانية بالقامرة		7	Graph Theory 7.1 Primitives 7.2 Articulation Points & Bridges 7.3 SCC 7.4 2-SAT 7.5 Edmonds-Karp Max Flow 7.6 Dinic's Max Flow 7.7 Min-Cost Max Flow 7.8 Euler Cycles 7.9 Maximum Matching	5 6 6 6 6 6 7 7
Contents				7.10 HL Decomposition	7
1	2D Geometry	2		7.11 Modelling Inequalities	7 7
	1.1 Primitives	2		7.13 Bellman Ford	7
	1.2 Intersections	2		7.14 Stable Marriage	7
	1.3 Circle Generation	2		7.15 Maximum Assignment	7
	1.4 Heron Triangle Area	2	6	Liu and Dunamananian	7
	1.5 Polygon Centroid	2	8	Linear Programming 8.1 Simplex	7 7
	1.6 Point In Polygon	2		6.1 Simplex	'
	1.7 Convex Hull	2	9	Number Theory	8
	1.8 Line Segment Set Intersection	2		9.1 Extended GCD	8
	1.9 Voronoi Diagrams	3		9.2 Modular Inverse	8
_	an G			9.3 Modular Linear Equation9.4 Linear Diophantine Equation	8 8
2	3D Geometry	3		9.5 Modular Powers	8
	2.1 Primitives			9.6 Sieve of Eratosthenes	8
	2.2 Convex Hull			9.7 Primality Testing & Factoring	8
	2.3 Great Circle Distance	3		9.8 Euler Phi	8
3	Combinatorics	3		9.9 Chinese Remainder	8
•	3.1 Basics			9.10 Discerete Logarithm	8
	3.2 Permutation (Un)Ranking			9.12 Pollard Rho	9
	3.3 Combination (Un)Ranking				
	(), 0		10	O Search	9
4	Data Structures	4		10.1 Binary Search	9
	4.1 Palindromic Tree	4		10.2 Ternary Search	9
	4.2 Treap	4	11	1 Strings	9
	4.3 Sparse Array	4		11.1 Aho Corasick	9
	4.4 Skip Lists			11.2 Hashing	9
	4.5 BIT + Search \dots			11.3 Z-Algorithm	9
	4.6 Segment Tree + Lazy Propagation			11.4 KMP + Periods	9
	4.7 Weighted Union Disjoint Sets	5		11.6 Suffix Array	9
5	Game Theory	5	19	2 Misc	9
	5.1 Nim Game		14	2 IVIISC	9
	5.2 Grundy Numbers	5	13	3 Last Page	10
	5.3 General Josephus Problem	5			
6	General Mathematics	5			
	6.1 Inclusion-Exclusion Patterns	5			
	6.2 Determinant				
	6.3 Gaussian Elimination	5			
	6.4 Fast Fourier-Transform	5			
	6.5 Misc. Formulas	5			

Bool Shift!

1 2D Geometry

```
16 // Algorithm
17 // Algorithm
18 // Algorithm
19 // From 3 Lines
    1.1 Primitives
 1 typedef complex<double> point;
                                                                         20 // Algorithm
21 // Algorithm
   struct circle {
  point c; double r;
                                                                         22 // Algorithm
23 // Algorithm
      circle(point c, double r):c(c),r(r){}
     circle(){}
                                                                         24 // Algorithm
 6 };
   double cross(const point &a, const point &b) {
 7
                                                                             1.4 Heron Triangle Area
     return imag(conj(a)*b);
 9 }
                                                                          1 // Formula
double dot(const point &a, const point &b) {
                                                                          2 // Formula
11
      return real(conj(a)*b);
                                                                          3 // Formula
12
                                                                             1.5 Polygon Centroid
    1.2 Intersections
                                                                          1 for(int i = 1; i < n-1; i++) {</pre>
 1 // Line - Line
                                                                               pt ai = pts[i] - pts[i-1],
  ib = pts[i+1] - pts[i];
   // Algorithm
 3 // Algorithm
                                                                               area += (conj(ai)*ib).imag();
 4 // Algorithm
 5 // Algorithm
 6 // Algorithm
                                                                             1.6 Point In Polygon
 7 // Line - Segment
 8 // Algorithm
 9 // Algorithm
                                                                          1 // Algorithm
                                                                          2 // Algorithm
10 // Algorithm
11 // Algorithm
                                                                          3 // Algorithm
                                                                          4 // Algorithm
12 // Algorithm
                                                                          5 // Algorithm
6 // Algorithm
13 // Segment - Segment
14 // Algorithm
15 // Algorithm
                                                                          7 // Algorithm
16 // Algorithm
17 // Algorithm
                                                                          8 // Algorithm
                                                                          9 // Algorithm
                                                                         10 // Algorithm
18 // Algorithm
19 // Circle - Line
                                                                         11 // Algorithm
20 // Algorithm
21 // Algorithm
22 // Algorithm
23 // Algorithm
                                                                         12 // Algorithm
                                                                         13 // Algorithm
14 // Algorithm
                                                                         15 // Algorithm
24 // Algorithm
25 // Circle - Segment
                                                                         16 // Algorithm
                                                                         17 // Algorithm
26 // Algorithm
27 // Algorithm
                                                                         18 // Algorithm
                                                                         19 // Algorithm
                                                                         20 // Algorithm
28 // Algorithm
29 // Algorithm
30 // Algorithm
31 // Circle - Circle
                                                                             1.7 Convex Hull
32 // Algorithm
33 // Algorithm
                                                                          1 // Algorithm
                                                                          2 // Algorithm
34 // Algorithm
35 // Algorithm
                                                                          3 // Algorithm
4 // Algorithm
36 // Algorithm
                                                                          5 // Algorithm
                                                                          6 // Algorithm
37 // Line - Point
                                                                          7 // Algorithm
8 // Algorithm
38 // Algorithm
39 // Algorithm
40 // Algorithm
                                                                          9 // Algorithm
                                                                         10 // Algorithm
41 // Algorithm
                                                                         11 // Algorithm
12 // Algorithm
42 // Algorithm
43 // Segment - Point
44 // Algorithm
                                                                         13 // Algorithm
45 // Algorithm
                                                                         14 // Algorithm
46 // Algorithm
                                                                         15 // Algorithm
47 // Algorithm
                                                                         16 // Algorithm
                                                                         17 // Algorithm
48 // Algorithm
                                                                          18 // Algorithm
                                                                          19 // Algorithm
   1.3 Circle Generation
                                                                          20 // Algorithm
 1 // From 3 Points
 2 // Algorithm
3 // Algorithm
                                                                             1.8 Line Segment Set Intersection
 4 // Algorithm
5 // Algorithm
                                                                          1 // Algorithm
                                                                          2 // Algorithm
                                                                          3 // Algorithm
4 // Algorithm
 6 // Algorithm
 7 // From 1 Line 2 Points
                                                                          5 // Algorithm
6 // Algorithm
 8 // Algorithm
 9 // Algorithm
                                                                          7 // Algorithm
8 // Algorithm
10 // Algorithm
11 // Algorithm
12 // Algorithm
                                                                          9 // Algorithm
                                                                          10 // Algorithm
13 // From 2 Lines 1 Point
14 // Algorithm
                                                                          11 // Algorithm
15 // Algorithm
                                                                         12 // Algorithm
```

```
13 // Algorithm
                                                                    3 // Algorithm
14 // Algorithm
15 // Algorithm
                                                                    4 // Algorithm
5 // Algorithm
16 // Algorithm
                                                                    6 // Algorithm
17 // Algorithm
18 // Algorithm
                                                                    7 // Algorithm
8 // Algorithm
19 // Algorithm
                                                                    9 // Algorithm
                                                                   10 // Algorithm
20 // Algorithm
                                                                   11 // Algorithm
12 // Algorithm
21 // Algorithm
22 // Algorithm
23 // Algorithm
                                                                   13 // Algorithm
                                                                   14 // Algorithm
24 // Algorithm
                                                                   15 // Algorithm
16 // Algorithm
25 // Algorithm
26 // Algorithm
                                                                   17 // Algorithm
18 // Algorithm
27 // Algorithm
28 // Algorithm
29 // Algorithm
                                                                   19 // Algorithm
30 // Algorithm
                                                                   20 // Algorithm
31 // Algorithm
32 // Algorithm
                                                                      2.3 Great Circle Distance
33 // Algorithm
34 // Algorithm
                                                                    2 // Code
3 // Code
35 // Algorithm
36 // Algorithm
                                                                    4 // Code
5 // Code
37 // Algorithm
38 // Algorithm
39 // Algorithm
                                                                    6 // Code
                                                                    7 // Code
40 // Algorithm
41 // Algorithm
                                                                    8 // Code
42 // Algorithm
                                                                    9 // Code
43 // Algorithm
                                                                   10 // Code
44 // Algorithm
45 // Algorithm
                                                                      3 Combinatorics
46 // Algorithm
                                                                      3.1 Basics
47 // Algorithm
48 // Algorithm
                                                                      // catalan numbers
49 // Algorithm
                                                                      long long C(int n) {
                                                                        return (C(n-1)*2*n*(2*n-1))/(n*(n+1));
return NCR(2*n, n) - NCR(2*n, n+1);
   1.9 Voronoi Diagrams
                                                                        return NCR(2*n, n)/(n+1);
1 // Algorithm
                                                                    6 }
2 // Algorithm
3 // Algorithm
                                                                    7 // derangements
                                                                    8
                                                                      long long D(int n) {
4 // Algorithm
                                                                    9
                                                                        return n*D(n-1) + pow(-1, n);
5 // Algorithm
                                                                   10
                                                                        return (n-1) * (D(n-1) + D(n-2));
6 // Algorithm
                                                                   11 }
7 // Algorithm
                                                                   12 // iterate over all subsets with < m elements
8 // Algorithm
                                                                   13 for (int i = 0; i < (1<<n); i=Integer.bitCount(i) < m ? i</pre>
9 // Algorithm
                                                                           +1 : (i|(i-1))+1)
10 // Algorithm
                                                                   14 // iterate over all the subsets
11 // Algorithm
                                                                   15 for (int i=0; i < (1<<n); i++)
12 // Algorithm
                                                                        // iterate over all the subsets of the i-th subset
13 // Algorithm
                                                                      for(int i2 = i; i2 > 0; i2 = (i2-1) & i)
14 // Algorithm
15 // Algorithm
                                                                      3.2 Permutation (Un)Ranking
16 // Algorithm
17 // Algorithm
                                                                    1 // Algorithm
18 // Algorithm
                                                                      // Algorithm
19 // Algorithm
                                                                      // Algorithm
20 // Algorithm
                                                                    4 // Algorithm
                                                                    5 // Algorithm
   2 3D Geometry
                                                                    6 // Algorithm
   2.1 Primitives
                                                                    7 // Algorithm
                                                                    8 // Algorithm
                                                                    9 // Algorithm
1 // Code
                                                                   10 // Algorithm
2 // Code
                                                                   11 // Algorithm
3 // Code
                                                                   12 // Algorithm
4 // Code
5 // Code
                                                                   13 // Algorithm
                                                                   14 // Algorithm
6 // Code
7 // Code
8 // Code
                                                                   15 // Algorithm
                                                                   16 // Algorithm
9 // Code
                                                                   17 // Algorithm
10 // Code
11 // Code
                                                                   18 // Algorithm
19 // Algorithm
12 // Code
                                                                   20 // Algorithm
13 // Code
14 // Code
                                                                      3.3 Combination (Un)Ranking
15 // Code
                                                                    1 // Algorithm
   2.2 Convex Hull
                                                                      // Algorithm
                                                                    3 // Algorithm
 1 // Algorithm
                                                                      // Algorithm
 2 // Algorithm
                                                                    5 // Algorithm
```

```
10 // Code
11 // Code
12 // Code
13 // Code
 6 // Algorithm
7 // Algorithm
8 // Algorithm
9 // Algorithm
10 // Algorithm
                                                                      14 // Code
11 // Algorithm
                                                                       15 // Code
12 // Algorithm
                                                                          4.4 Skip Lists
13 // Algorithm
14 // Algorithm
                                                                        1 // Code
15 // Algorithm
                                                                       2 // Code
3 // Code
16 // Algorithm
17 // Algorithm
                                                                       4 // Code
5 // Code
18 // Algorithm
19 // Algorithm
                                                                        6 // Code
7 // Code
20 // Algorithm
                                                                       8 // Code
9 // Code
   4 Data Structures
   4.1 Palindromic Tree
                                                                       10 // Code
11 // Code
   // Algorithm
                                                                      12 // Code
13 // Code
   // Algorithm
   // Algorithm
                                                                      14 // Code
15 // Code
   // Algorithm
5 // Algorithm
                                                                      16 // Code
17 // Code
6 // Algorithm
7 // Algorithm
                                                                      18 // Code
19 // Code
8 // Algorithm
9 // Algorithm
                                                                      20 // Code
21 // Code
10 // Algorithm
11 // Algorithm
                                                                      22 // Code
12 // Algorithm
13 // Algorithm
                                                                          4.5 BIT + Search
14 // Algorithm
15 // Algorithm
16 // Algorithm
                                                                        1 // Algorithm
                                                                          // Algorithm
17 // Algorithm
18 // Algorithm
                                                                          // Algorithm
                                                                          // Algorithm
19 // Algorithm
                                                                          // Algorithm
20 // Algorithm
                                                                          // Algorithm
                                                                          // Algorithm
   4.2 Treap
                                                                          // Algorithm
                                                                          // Algorithm
1 // Algorithm
                                                                       10 // Algorithm
   // Algorithm
                                                                       11 // Algorithm
3 // Algorithm
                                                                       12 // Algorithm
 4 // Algorithm
                                                                       13 // Algorithm
5 // Algorithm
                                                                       14 // Algorithm
 6 // Algorithm
                                                                      15 // Algorithm
7 // Algorithm
                                                                      16 // Algorithm
 8 // Algorithm
                                                                      17 // Algorithm
9 // Algorithm
                                                                       18 // Algorithm
10 // Algorithm
                                                                      19 // Algorithm
11 // Algorithm
                                                                       20 // Algorithm
12 // Algorithm
13 // Algorithm
                                                                          4.6 Segment Tree + Lazy Propagation
14 // Algorithm
15 // Algorithm
                                                                          // Algorithm
16 // Algorithm
                                                                          // Algorithm
17 // Algorithm
                                                                          // Algorithm
                                                                        3
18 // Algorithm
                                                                        4 // Algorithm
19 // Algorithm
                                                                          // Algorithm
20 // Algorithm
                                                                          // Algorithm
21 // Algorithm
                                                                          // Algorithm
22 // Algorithm
                                                                        8 // Algorithm
                                                                          // Algorithm
23 // Algorithm
                                                                       10 // Algorithm
24 // Algorithm
25 // Algorithm
                                                                      11 // Algorithm
12 // Algorithm
26 // Algorithm
                                                                      13 // Algorithm
14 // Algorithm
27 // Algorithm
28 // Algorithm
   // Algorithm
                                                                      15 // Algorithm
16 // Algorithm
29
30 // Algorithm
                                                                      17 // Algorithm
18 // Algorithm
   4.3 Sparse Array
                                                                       19 // Algorithm
1 // Code
                                                                       20 // Algorithm
2 // Code
3 // Code
                                                                      21 // Algorithm
22 // Algorithm
4 // Code
5 // Code
                                                                      23 // Algorithm
24 // Algorithm
6 // Code
                                                                          // Algorithm
                                                                       26 // Algorithm
7 // Code
   // Code
                                                                          // Algorithm
 9 // Code
                                                                          // Algorithm
```

```
29 // Algorithm
                                                                    9 // Algorithm
                                                                   10 // Algorithm
11 // Algorithm
12 // Algorithm
30 // Algorithm
   4.7 Weighted Union Disjoint Sets
                                                                   13 // Algorithm
14 // Algorithm
 1 // Algorithm
2 // Algorithm
                                                                   15 // Algorithm
3 // Algorithm
 4 // Algorithm
                                                                       6.3 Gaussian Elimination
5 // Algorithm
 6 // Algorithm
                                                                    double* GaussianElimination(int N, double **mat) {
                                                                         int i, j, k, L; double t;
for (i = 0; i < N - 1; i++) {</pre>
7 // Algorithm
 8 // Algorithm
9 // Algorithm
                                                                           L = i;
                                                                           for (j = i + 1; j < N; j++)
  if (fabs(mat[j][i]) > fabs(mat[L][i]))
10 // Algorithm
11 // Algorithm
12 // Algorithm
                                                                               L = j;
13 // Algorithm
                                                                           for (k = i; k \le N; k++)
                                                                           swap(mat[i][k], mat[L][k]);
for (j = i + 1; j < N; j++)
  for (k = N; k >= i; k--)
14 // Algorithm
                                                                    9
15 // Algorithm
                                                                    10
                                                                    11
                                                                               mat[j][k] -= (mat[i][k] * mat[j][i]) / mat[i][i];
      Game Theory
   5
                                                                    12
                                                                    13
   5.1 Nim Game
                                                                         double *res = new double[N];
                                                                    14
                                                                         for (j = N - 1; j >= 0; j--) {
for (t = 0.0, k = j + 1; k < N; k++)
                                                                    15
 1 // Algorithm
                                                                    16
   // Algorithm
                                                                           t += mat[j][k] * res[k];
                                                                   17
3 // Algorithm
                                                                           res[j] = (mat[j][N] - t) / mat[j][j];
                                                                   18
4 // Algorithm
                                                                   19
5 // Algorithm
                                                                   20
                                                                         return res;
6 // Algorithm
                                                                   21 }
7 // Algorithm
8 // Algorithm
                                                                       6.4 Fast Fourier-Transform
9 // Algorithm
10 // Algorithm
                                                                    1 // Algorithm
                                                                    2 // Algorithm
3 // Algorithm
   5.2 Grundy Numbers
                                                                    4 // Algorithm
5 // Algorithm
 1 // Algorithm
2 // Algorithm
                                                                    6 // Algorithm
7 // Algorithm
3 // Algorithm
4 // Algorithm
                                                                    8 // Algorithm
9 // Algorithm
5 // Algorithm
6 // Algorithm
                                                                    10 // Algorithm
7 // Algorithm
8 // Algorithm
                                                                       6.5 Misc. Formulas
9 // Algorithm
10 // Algorithm
                                                                    1 // Algorithm
11 // Algorithm
12 // Algorithm
                                                                    2 // Algorithm
                                                                    3 // Algorithm
13 // Algorithm
                                                                    4 // Algorithm
14 // Algorithm
                                                                    5 // Algorithm
15 // Algorithm
                                                                    6 // Algorithm
16 // Algorithm
                                                                    7 // Algorithm
17 // Algorithm
                                                                    8 // Algorithm
18 // Algorithm
                                                                    9 // Algorithm
19 // Algorithm
                                                                    10 // Algorithm
20 // Algorithm
                                                                    11 // Algorithm
                                                                    12 // Algorithm
   5.3 General Josephus Problem
                                                                   13 // Algorithm
   6 General Mathematics
                                                                   14 // Algorithm
                                                                   15 // Algorithm
   6.1 Inclusion-Exclusion Patterns
                                                                   16 // Algorithm
                                                                    17 // Algorithm
1 // Algorithm
                                                                   18 // Algorithm
   // Algorithm
                                                                   19 // Algorithm
3 // Algorithm
                                                                   20 // Algorithm
4 // Algorithm
                                                                   21 // Algorithm
5 // Algorithm
                                                                   22 // Algorithm
6 // Algorithm
                                                                   23 // Algorithm
7 // Algorithm
                                                                   24 // Algorithm
8 // Algorithm
                                                                   25 // Algorithm
   // Algorithm
                                                                   26 // Algorithm
27 // Algorithm
10 // Algorithm
                                                                   28 // Algorithm
   6.2 Determinant
                                                                   29 // Algorithm
                                                                    30 // Algorithm
 1 // Algorithm
2 // Algorithm
                                                                          Graph Theory
3 // Algorithm
                                                                       7.1 Primitives
4 // Algorithm
5 // Algorithm
 6 // Algorithm
                                                                    1 // Algorithm
   // Algorithm
                                                                       // Algorithm
 8 // Algorithm
                                                                       // Algorithm
```

```
7.4 2-SAT
 4 // Algorithm
5 // Algorithm
6 // Algorithm
                                                                         1 // Algorithm
                                                                           // Algorithm
7 // Algorithm
                                                                           // Algorithm
8 // Algorithm
                                                                         4 // Algorithm
9 // Algorithm
                                                                           // Algorithm
10 // Algorithm
                                                                         6 // Algorithm
11 // Algorithm
                                                                           // Algorithm
12 // Algorithm
                                                                         8 // Algorithm
13 // Algorithm
                                                                       9 // Algorithm
10 // Algorithm
14 // Algorithm
15 // Algorithm
                                                                       11 // Algorithm
12 // Algorithm
   7.2 Articulation Points & Bridges
                                                                       13 // Algorithm
14 // Algorithm
1 // Algorithm
                                                                       15 // Algorithm
16 // Algorithm
   // Algorithm
3 // Algorithm
                                                                       17 // Algorithm
18 // Algorithm
 4 // Algorithm
5 // Algorithm
 6 // Algorithm
                                                                       19 // Algorithm
7 // Algorithm
                                                                       20 // Algorithm
 8 // Algorithm
                                                                           7.5 Edmonds-Karp Max Flow
9 // Algorithm
10 // Algorithm
11 // Algorithm
                                                                         1 // Algorithm
                                                                        2 // Algorithm
3 // Algorithm
12 // Algorithm
13 // Algorithm
                                                                         4 // Algorithm
5 // Algorithm
14 // Algorithm
15 // Algorithm
16 // Algorithm
                                                                           // Algorithm
17 // Algorithm
                                                                           // Algorithm
18 // Algorithm
                                                                           // Algorithm
19 // Algorithm
                                                                         9 // Algorithm
                                                                        10 // Algorithm
11 // Algorithm
20 // Algorithm
                                                                       12 // Algorithm
13 // Algorithm
   7.3 SCC
   #define MAXN 100
                                                                       14 // Algorithm
                                                                       15 // Algorithm
                                                                       16 // Algorithm
                                                                       17 // Algorithm
   vector< vector<int> > adj;
                                                                       18 // Algorithm
   int num[MAXN], low[MAXN];
                                                                       19 // Algorithm
   bool vis[MAXN];
                                                                       20 // Algorithm
   vector<int> S;
   vector< vector<int> > SCCs;
                                                                           7.6 Dinic's Max Flow
10
   int dfsNumber;
                                                                         1 // Algorithm
11
   void tarjanSCC(int u) {
                                                                         2 // Algorithm
12
     low[u] = num[u] = dfsNumber ++;
                                                                         3 // Algorithm
13
     S.push_back(u);
                                                                         4 // Algorithm
14
     vis[u] = 1;
                                                                         5 // Algorithm
15
     for(int i=0; i<adj[u].size(); i++) {</pre>
                                                                         6 // Algorithm
16
        int v = adj[u][i];
if(num[v] == -1)
                                                                         7 // Algorithm
17
                                                                         8 // Algorithm
18
          tarjanSCC(v);
                                                                         9 // Algorithm
19
        if(vis[v] == 1)
  low[u] = min(low[u], low[v]);
                                                                       10 // Algorithm
20
                                                                       11 // Algorithm
21
                                                                       12 // Algorithm
22
                                                                       13 // Algorithm
23
     if(low[u] == num[u]) {
        vector<int> SCC;
                                                                       14 // Algorithm
24
                                                                       15 // Algorithm
        while(1) {
  int v = S.back();
25
                                                                       16 // Algorithm
17 // Algorithm
26
          S.pop_back();
27
                                                                       18 // Algorithm
19 // Algorithm
          vis[v] = 0;
28
          SCC.push_back(v);
29
          if(u == v)
                                                                       20 // Algorithm
30
31
            break;
32
                                                                           7.7 Min-Cost Max Flow
        SCCs.push_back(SCC);
33
                                                                        1 // Algorithm
2 // Algorithm
34
35
                                                                        3 // Algorithm
4 // Algorithm
36
   void findSCC() {
37
38
     dfsNumber = 0;
                                                                         5 // Algorithm
     memset(vis, 0, sizeof(vis));
memset(num, -1, sizeof(num));
                                                                         6 // Algorithm
39
                                                                        7 // Algorithm
8 // Algorithm
40
41
     memset(low, 0, sizeof(low));
                                                                       9 // Algorithm
10 // Algorithm
     S.clear();
43
     SCCs.clear();
                                                                       11 // Algorithm
12 // Algorithm
     for(int i=0; i<N; i++)</pre>
44
45
       if(num[i] == -1)
          tarjanSCC(i);
                                                                       13 // Algorithm
47
                                                                       14 // Algorithm
```

```
15 // Algorithm
                                                                    7 // Algorithm
16 // Algorithm
17 // Algorithm
                                                                    8 // Algorithm
9 // Algorithm
18 // Algorithm
                                                                   10 // Algorithm
19 // Algorithm
                                                                      7.12 Max Flow Tricks
20 // Algorithm
   7.8 Euler Cycles
                                                                    1 // Algorithm
                                                                      // Algorithm
// Algorithm
 1 // Algorithm
   // Algorithm
                                                                    4 // Algorithm
                                                                    5 // Algorithm
3 // Algorithm
                                                                      // Algorithm
4 // Algorithm
5 // Algorithm
6 // Algorithm
                                                                      // Algorithm
                                                                    8 // Algorithm
7 // Algorithm
8 // Algorithm
                                                                    9 // Algorithm
                                                                   10 // Algorithm
11 // Algorithm
9 // Algorithm
10 // Algorithm
11 // Algorithm
                                                                   12 // Algorithm
                                                                   13 // Algorithm
                                                                   14 // Algorithm
12 // Algorithm
                                                                   15 // Algorithm
13 // Algorithm
14 // Algorithm
                                                                      7.13 Bellman Ford
15 // Algorithm
16 // Algorithm
                                                                      // Algorithm
17 // Algorithm
                                                                      // Algorithm
18 // Algorithm
                                                                      // Algorithm
19 // Algorithm
                                                                      // Algorithm
20 // Algorithm
                                                                      // Algorithm
                                                                    6 // Algorithm
   7.9 Maximum Matching
                                                                      // Algorithm
                                                                      // Algorithm
 1 // Algorithm
                                                                      // Algorithm
2 // Algorithm
3 // Algorithm
                                                                   10 // Algorithm
4 // Algorithm
                                                                      7.14 Stable Marriage
5 // Algorithm
6 // Algorithm
                                                                    1 // Algorithm
7 // Algorithm
8 // Algorithm
                                                                    2 // Algorithm
3 // Algorithm
9 // Algorithm
                                                                    4 // Algorithm
10 // Algorithm
                                                                    5 // Algorithm
11 // Algorithm
                                                                    6 // Algorithm
7 // Algorithm
12 // Algorithm
13 // Algorithm
                                                                    8 // Algorithm
9 // Algorithm
14 // Algorithm
15 // Algorithm
                                                                   10 // Algorithm
16 // Algorithm
                                                                   11 // Algorithm
17 // Algorithm
                                                                   12 // Algorithm
18 // Algorithm
                                                                   13 // Algorithm
19 // Algorithm
                                                                   14 // Algorithm
20 // Algorithm
                                                                   15 // Algorithm
   7.10 HL Decomposition
                                                                      7.15 Maximum Assignment
 1 // Algorithm
                                                                    1 // Algorithm
   // Algorithm
                                                                      // Algorithm
3 // Algorithm
                                                                    3 // Algorithm
4 // Algorithm
                                                                    4 // Algorithm
5 // Algorithm
                                                                    5 // Algorithm
6 // Algorithm
                                                                    6 // Algorithm
7 // Algorithm
                                                                    7 // Algorithm
8 // Algorithm
                                                                    8 // Algorithm
9 // Algorithm
                                                                    9 // Algorithm
10 // Algorithm
                                                                   10 // Algorithm
11 // Algorithm
                                                                   11 // Algorithm
12 // Algorithm
                                                                   12 // Algorithm
13 // Algorithm
                                                                   13 // Algorithm
14 // Algorithm
14 // Algorithm
15 // Algorithm
                                                                   15 // Algorithm
16 // Algorithm
17 // Algorithm
                                                                      8 Linear Programming
   // Algorithm
19 // Algorithm
                                                                      8.1 Simplex
20 // Algorithm
                                                                    1 // Algorithm
   7.11 Modelling Inequalities
                                                                      // Algorithm
                                                                      // Algorithm
                                                                    4 // Algorithm
5 // Algorithm
 1 // Algorithm
2 // Algorithm
3 // Algorithm
                                                                      // Algorithm
                                                                      // Algorithm
 4 // Algorithm
   // Algorithm
                                                                      // Algorithm
 6 // Algorithm
                                                                      // Algorithm
```

```
10 // Algorithm
                                                                  7 // Algorithm
11 // Algorithm
12 // Algorithm
                                                                  8 // Algorithm
9 // Algorithm
13 // Algorithm
                                                                 10 // Algorithm
14 // Algorithm
15 // Algorithm
                                                                    9.6 Sieve of Eratosthenes
16 // Algorithm
                                                                  1 // Algorithm
17 // Algorithm
                                                                  2 // Algorithm
3 // Algorithm
18 // Algorithm
19 // Algorithm
20 // Algorithm
                                                                  4 // Algorithm
                                                                  5 // Algorithm
21 // Algorithm
22 // Algorithm
                                                                  6 // Algorithm
23 // Algorithm
                                                                  7 // Algorithm
                                                                  8 // Algorithm
9 // Algorithm
24 // Algorithm
25 // Algorithm
26 // Algorithm
                                                                 10 // Algorithm
27 // Algorithm
                                                                    9.7 Primality Testing & Factoring
28 // Algorithm
29 // Algorithm
30 // Algorithm
                                                                  1 // Algorithm
31 // Algorithm
                                                                    // Algorithm
                                                                    // Algorithm
// Algorithm
   9 Number Theory
                                                                    // Algorithm
   9.1 Extended GCD
                                                                  6 // Algorithm
                                                                    // Algorithm
1 long long gcd( long long a, long long b )
                                                                  8 // Algorithm
   { return(b == 0 ? a : gcd(b, a % b ) ); }
//USED BY: egcd, msolve, inverse, ldioph
                                                                    // Algorithm
                                                                 10 // Algorithm
  template< class Int > struct Triple {
    Int d, x, y;
                                                                    9.8 Euler Phi
    \label{eq:triple} \texttt{Triple}(\texttt{Int}\ q,\ \texttt{Int}\ w,\ \texttt{Int}\ e): d(q)\,,\ x(w)\,,\ y(e)\,\{\}
7 };
                                                                  1 // Algorithm
   //USED BY: msolve, inverse, ldioph
                                                                    // Algorithm
9 template < class Int > Triple < Int > egcd( Int a, Int b ) { 3 // Algorithm
    if(!b) return Triple< Int > (a, Int(1), Int(0)); 4 // Algorithm
Triple< Int > q = egcd(b, a % b);
10
11
     12
                                                                  6 // Algorithm
13
                                                                  7 // Algorithm
                                                                    // Algorithm
   9.2 Modular Inverse
                                                                    // Algorithm
                                                                 10 // Algorithm
1 //solves ax = 1 (mod n).
  template< class Int > Int inverse( Int a, Int n ) {
                                                                    9.9 Chinese Remainder
3
       Triple< Int > t = egcd( a, n );
       if( t.d > Int( 1 ) ) return Int( 0 );
                                                                  1 // Algorithm
       Int r = t.x % n;
                                                                  2 // Algorithm
       return( r < Int( 0 ) ? r + n : r );</pre>
                                                                  3 // Algorithm
                                                                  4 // Algorithm
                                                                  5 // Algorithm
   9.3 Modular Linear Equation
                                                                  6 // Algorithm
                                                                  7 // Algorithm
1 // Algorithm
                                                                  8 // Algorithm
2 // Algorithm
                                                                  9 // Algorithm
3 // Algorithm
                                                                 10 // Algorithm
4 // Algorithm
5 // Algorithm
                                                                    9.10 Discerete Logarithm
6 // Algorithm
7 // Algorithm
                                                                  1 // Algorithm
  // Algorithm
                                                                  2 // Algorithm
3 // Algorithm
   // Algorithm
10 // Algorithm
                                                                  4 // Algorithm
                                                                  5 // Algorithm
6 // Algorithm
   9.4 Linear Diophantine Equation
                                                                    // Algorithm
1 // Algorithm
                                                                  8 // Algorithm
2 // Algorithm
3 // Algorithm
                                                                    // Algorithm
                                                                 10 // Algorithm
4 // Algorithm
5 // Algorithm
                                                                    9.11 Tortoise & Hare
6 // Algorithm
7 // Algorithm
8 // Algorithm
                                                                     // mu = start of cycle, lambda = cycle length
                                                                     ii floyd(int x0) {
9 // Algorithm
                                                                      int tortoise = f(x0), hare = f(f(x0));
10 // Algorithm
                                                                      while(tortoise != hare)
                                                                        tortoise = f(tortoise), hare = f(f(hare));
   9.5 Modular Powers
                                                                      int mu = 0; hare = x0;
                                                                       while(tortoise != hare)
 1 // Algorithm
                                                                        tortoise = f(tortoise), hare = f(hare), mu++;
                                                                      int lambda = 1; hare = f(tortoise);
2 // Algorithm
                                                                  9
3 // Algorithm
                                                                       while(tortoise != hare)
                                                                 10
4 // Algorithm
                                                                 11
                                                                        hare = f(hare), lambda++;
  // Algorithm
                                                                      return ii(mu, lambda);
 6 // Algorithm
```

```
9.12 Pollard Rho
                                                                    8 // Algorithm
                                                                  9 // Algorithm
10 // Algorithm
11 // Algorithm
   10 Search
   10.1 Binary Search
                                                                  12 // Algorithm
13 // Algorithm
1 // Algorithm
2 // Algorithm
3 // Algorithm
                                                                   14 // Algorithm
                                                                   15 // Algorithm
4 // Algorithm
                                                                      11.4 \text{ KMP} + \text{Periods}
5 // Algorithm
6 // Algorithm
                                                                    1 // Algorithm
7 // Algorithm
                                                                      // Algorithm
8 // Algorithm
                                                                    3 // Algorithm
9 // Algorithm
                                                                    4 // Algorithm
10 // Algorithm
                                                                    5 // Algorithm
11 // Algorithm
                                                                    6 // Algorithm
12 // Algorithm
                                                                    7 // Algorithm
13 // Algorithm
                                                                    8 // Algorithm
14 // Algorithm
                                                                    9 // Algorithm
15 // Algorithm
                                                                   10 // Algorithm
                                                                   11 // Algorithm
   10.2 Ternary Search
                                                                   12 // Algorithm
                                                                  13 // Algorithm
  long double min() {
                                                                  14 // Algorithm
       long double lo = -le6, hi = le6, res = 3e6;
                                                                  15 // Algorithm
       while (fabs (lo-hi) > EPS) {
3
           long double left = (hi-lo)/3 + lo, right = (2*(hi-
4
                                                                     11.5 Manacher
                10))/3 + 10;
           long double resL = F(left), resR = F(right);
                                                                    1 // Algorithm
           if(resL < resR)</pre>
6
                                                                    2 // Algorithm
               hi = right;
7
                                                                    3 // Algorithm
           else
                                                                    4 // Algorithm
              lo = left;
9
                                                                     // Algorithm
           res = min(res, min(resL, resR));
10
                                                                    6 // Algorithm
11
                                                                     // Algorithm
// Algorithm
12
       return res;
13
                                                                   9 // Algorithm
10 // Algorithm
   11 Strings
                                                                   11 // Algorithm
                                                                   12 // Algorithm
   11.1 Aho Corasick
                                                                   13 // Algorithm
                                                                   14 // Algorithm
1 // Algorithm
                                                                  15 // Algorithm
2 // Algorithm
                                                                   16 // Algorithm
3 // Algorithm
                                                                  17 // Algorithm
4 // Algorithm
                                                                   18 // Algorithm
5 // Algorithm
                                                                  19 // Algorithm
6 // Algorithm
                                                                   20 // Algorithm
7 // Algorithm
8 // Algorithm
                                                                   21 // Algorithm
9 // Algorithm
                                                                      11.6 Suffix Array
10 // Algorithm
11 // Algorithm
                                                                    1 // Algorithm
12 // Algorithm
                                                                    2 // Algorithm
13 // Algorithm
                                                                      // Algorithm
14 // Algorithm
                                                                    4 // Algorithm
15 // Algorithm
                                                                    5 // Algorithm
                                                                    6 // Algorithm
   11.2 Hashing
                                                                    7 // Algorithm
8 // Algorithm
1 // Algorithm
                                                                  9 // Algorithm
10 // Algorithm
2 // Algorithm
3 // Algorithm
                                                                  11 // Algorithm
12 // Algorithm
4 // Algorithm
5 // Algorithm
                                                                  13 // Algorithm
14 // Algorithm
6 // Algorithm
7 // Algorithm
                                                                  15 // Algorithm
8 // Algorithm
                                                                   16 // Algorithm
9 // Algorithm
                                                                  17 // Algorithm
18 // Algorithm
10 // Algorithm
11 // Algorithm
                                                                  19 // Algorithm
20 // Algorithm
12 // Algorithm
13 // Algorithm
                                                                   21 // Algorithm
14 // Algorithm
                                                                   22 // Algorithm
15 // Algorithm
                                                                   23 // Algorithm
                                                                   24 // Algorithm
   11.3 Z-Algorithm
                                                                   25 // Algorithm
                                                                   26 // Algorithm
1 // Algorithm
                                                                   27 // Algorithm
2 // Algorithm
                                                                   28 // Algorithm
3 // Algorithm
                                                                   29 // Algorithm
4 // Algorithm
                                                                   30 // Algorithm
5 // Algorithm
6 // Algorithm
                                                                      12
                                                                           \mathbf{Misc}
7 // Algorithm
```

13 Last Page

Cut this paper out. Use it wisely!

Problem	Tags
01 A	
02 B	
03 C	
04 D	
05 E	
06 F	
07 G	
08 H	
09 I	
10 J	
11 K	
12 L	
13 M	

Time	Meeting Description	Chk
030	All Problems Read. Write Tags.	
060	Ace Decided. Choose Coder.	
090	Decide & Order Solveable Problems	
120	Status Check	
150	Status Check	
180	Status Check	
210	Status Check	
240	Blind Hour. One Problem.	
270	Status Check	
300	Contest Ends	

Solving A Problem

Read the statement carefully.

Break the problem down into pieces.

Plan the solution to each piece.

Think of corner cases to solution.

Calculate Complexity.

Simplify the solution.

Write steps of solution on paper.

Estimate Coding time.

During The Contest

Stay **calm** and **focused** or you and your team mates won't make it

If you don't understand it, it doesn't mean it's hard. Tell your team mate.

READ THE STATEMENT AGAIN. TELL YOUR TEAMMATE!

Write **significant** tags! Think how all the topics might come in use!

Run over the index page. Maybe one of the topic titles will inspire a solution!

Common Bugs

Add eps to double before getting floor or round