

```

1 //CODECHEF - START66
2 //SPLITPAL
3 #ifdef WTSH
4     #include <wtsh.h>
5 #else
6     #include <bits/stdc++.h>
7     using namespace std;
8     #define dbg(...)
9 #endif
10
11 #define int long long
12 #define endl "\n"
13 #define sz(w) (int)(w.size())
14 using pii = pair<int, int>;
15
16 // ----- Input Checker Start -----
17
18 long long readInt(long long l, long long r, char endd)
19 {
20     long long x = 0;
21     int cnt = 0, fi = -1;
22     bool is_neg = false;
23     while(true)
24     {
25         char g = getchar();
26         if(g == '-')
27         {
28             assert(fi == -1);
29             is_neg = true;
30             continue;
31         }
32         if('0' <= g && g <= '9')
33         {
34             x *= 10;
35             x += g - '0';
36             if(cnt == 0)
37                 fi = g - '0';
38             cnt++;
39             assert(fi != 0 || cnt == 1);
40             assert(fi != 0 || is_neg == false);
41             assert(!(cnt > 19 || (cnt == 19 && fi > 1)));
42         }
43         else if(g == endd)
44         {
45             if(is_neg)
46                 x = -x;
47             if(!(l <= x && x <= r))
48             {
49                 cerr << "L: " << l << ", R: " << r << ", Value Found: " << x << '\n';
50                 assert(false);
51             }
52             return x;
53         }
54         else
55         {
56             assert(false);
57         }
58     }
59 }
60
61 string readString(int l, int r, char endd)
62 {
63     string ret = "";
64     int cnt = 0;
65     while(true)
66     {
67         char g = getchar();
68         assert(g != -1);
69         if(g == endd)

```

```

70         break;
71         cnt++;
72         ret += g;
73     }
74     assert(l <= cnt && cnt <= r);
75     return ret;
76 }
77
78 long long readIntSp(long long l, long long r) { return readInt(l, r, ' '); }
79 long long readIntLn(long long l, long long r) { return readInt(l, r, '\n'); }
80 string readStringSp(int l, int r) { return readString(l, r, ' '); }
81 string readStringLn(int l, int r) { return readString(l, r, '\n'); }
82 void readEOF() { assert(getchar() == EOF); }
83
84 vector<int> readVectorInt(int n, long long l, long long r)
85 {
86     vector<int> a(n);
87     for(int i = 0; i < n - 1; i++)
88         a[i] = readIntSp(l, r);
89     a[n - 1] = readIntLn(l, r);
90     return a;
91 }
92
93 // ----- Input Checker End -----
94
95 int sumN = 0;
96
97 void solve()
98 {
99     int n = readIntLn(1, 1e5);
100     sumN += n;
101     vector<int> a = readVectorInt(n, 1, 1e5);
102     int L = 0, R = n - 1, ans = 0;
103     while(L < R)
104     {
105         if(a[L] == a[R])
106             L++, R--;
107         else if(a[L] < a[R])
108             a[R] -= a[L], L++, ans++;
109         else
110             a[L] -= a[R], R--, ans++;
111     }
112     cout << ans << endl;
113 }
114
115 int32_t main()
116 {
117     ios::sync_with_stdio(0);
118     cin.tie(0);
119     int T = readIntLn(1, 1e5);
120     for(int tc = 1; tc <= T; tc++)
121     {
122         solve();
123     }
124     readEOF();
125     assert(sumN <= 2e5);
126     return 0;
127 }
128
129 //SPLITPAL-EDITOR
130 #include <bits/stdc++.h>
131 using namespace std;
132
133 int main() {
134     int t; cin >> t;
135     while (t--) {
136         int n; cin >> n;
137         deque<int> dq;
138         for (int i = 0; i < n; ++i) {

```

```

139         int x; cin >> x;
140         dq.push_back(x);
141     }
142     int ans = 0;
143     while (dq.size() > 1) {
144         int x = dq.front(), y = dq.back();
145         dq.pop_front(); dq.pop_back();
146         if (x == y) continue;
147         ++ans;
148         if (x < y) dq.push_back(y-x);
149         else dq.push_front(x-y);
150     }
151     cout << ans << '\n';
152 }
153 return 0;
154 }
155
156 //CNCTCT
157
158 #include <bits/stdc++.h>
159 using namespace std;
160 #define int long long
161 #define pb push_back
162 #define ff first
163 #define ss second
164 #define sz(x) (int)x.size()
165 #define all(x) x.begin(), x.end()
166 #define endl "\n"
167
168 int32_t main() {
169     ios::sync_with_stdio(false);
170     cin.tie(NULL);
171
172     int T = 1;
173     cin >> T;
174
175     while (T--) {
176         int n;
177         cin >> n;
178         vector<int> a(n);
179         for (auto &x : a)
180             cin >> x;
181         int ans = 0;
182         for (int k = 0; k < 30; ++k) {
183             vector<int> na;
184             int tt = 0, cnt = 0;
185             for (auto &x : a) {
186                 if (x & (1ll << k))
187                     tt |= x, cnt++;
188                 else
189                     na.pb(x);
190             }
191             if (cnt) {
192                 ans += (cnt - 1) * (1ll << k);
193                 na.pb(tt);
194                 swap(a, na);
195             }
196         }
197         if (sz(a) > 1)
198             cout << -1 << endl;
199         else
200             cout << ans << endl;
201     }
202     return 0;
203 }
204
205 //BSSORT
206 #ifdef WTSH

```

```

208     #include <wtsh.h>
209 #else
210     #include <bits/stdc++.h>
211     using namespace std;
212     #define dbg(...)
213 #endif
214
215 #define int long long
216 #define endl "\n"
217 #define sz(w) (int)(w.size())
218 using pii = pair<int, int>;
219
220 // ----- Input Checker Start -----
221
222 long long readInt(long long l, long long r, char endd)
223 {
224     long long x = 0;
225     int cnt = 0, fi = -1;
226     bool is_neg = false;
227     while(true)
228     {
229         char g = getchar();
230         if(g == '-')
231         {
232             assert(fi == -1);
233             is_neg = true;
234             continue;
235         }
236         if('0' <= g && g <= '9')
237         {
238             x *= 10;
239             x += g - '0';
240             if(cnt == 0)
241                 fi = g - '0';
242             cnt++;
243             assert(fi != 0 || cnt == 1);
244             assert(fi != 0 || is_neg == false);
245             assert(!(cnt > 19 || (cnt == 19 && fi > 1)));
246         }
247         else if(g == endd)
248         {
249             if(is_neg)
250                 x = -x;
251             if(!(l <= x && x <= r))
252             {
253                 cerr << "L: " << l << ", R: " << r << ", Value Found: " << x << '\n';
254                 assert(false);
255             }
256             return x;
257         }
258         else
259         {
260             assert(false);
261         }
262     }
263 }
264
265 string readString(int l, int r, char endd)
266 {
267     string ret = "";
268     int cnt = 0;
269     while(true)
270     {
271         char g = getchar();
272         assert(g != -1);
273         if(g == endd)
274             break;
275         cnt++;
276         ret += g;

```

```

277     }
278     assert(l <= cnt && cnt <= r);
279     return ret;
280 }
281
282 long long readIntSp(long long l, long long r) { return readInt(l, r, ' '); }
283 long long readIntLn(long long l, long long r) { return readInt(l, r, '\n'); }
284 string readStringSp(int l, int r) { return readString(l, r, ' '); }
285 string readStringLn(int l, int r) { return readString(l, r, '\n'); }
286 void readEOF() { assert(getchar() == EOF); }
287
288 vector<int> readVectorInt(int n, long long l, long long r)
289 {
290     vector<int> a(n);
291     for(int i = 0; i < n - 1; i++)
292         a[i] = readIntSp(l, r);
293     a[n - 1] = readIntLn(l, r);
294     return a;
295 }
296
297 // ----- Input Checker End -----
298
299 int sumN = 0;
300
301 void solve()
302 {
303     int n = readIntLn(1, 1e5);
304     sumN += n;
305     string s = readStringLn(n, n);
306     assert(*max_element(s.begin(), s.end()) <= '1');
307     assert(*min_element(s.begin(), s.end()) >= '0');
308     array<vector<int>, 2> occ{};
309     for(int i = 0; i < n; i++)
310         occ[s[i] - '0'].push_back(i);
311     int ans = 0;
312     for(int i = 0; i < n; i++)
313     {
314         int cur = i - 1;
315         for(int x: {1, 1, 0, 0})
316         {
317             auto it = upper_bound(occ[x].begin(), occ[x].end(), cur);
318             if(it == occ[x].end())
319             {
320                 cur = n;
321                 break;
322             }
323             cur = *it;
324         }
325         ans += cur - i;
326     }
327     cout << ans << endl;
328 }
329
330 int32_t main()
331 {
332     ios::sync_with_stdio(0);
333     cin.tie(0);
334     int T = readIntLn(1, 1e5);
335     for(int tc = 1; tc <= T; tc++)
336     {
337         solve();
338     }
339     readEOF();
340     assert(sumN <= 5e5);
341     return 0;
342 }
343
344 //BIN_BAT
345 #include <bits/stdc++.h>

```

```

346 using namespace std;
347
348 int main() {
349
350     ios_base::sync_with_stdio(false);
351     cin.tie(NULL);
352     cout.tie(NULL);
353
354     int t;
355     cin>>t;
356     while(t--){
357         int n,a,b;
358         cin>>n>>a>>b;
359         int k = log2(n);
360         int ans = (a*k)+(b*(k-1));
361         cout<<ans<<"\n";
362     }
363
364     return 0;
365 }
366
367 //CHEFSPICE
368 #include <bits/stdc++.h>
369
370 using namespace std;
371
372 int main()
373 {
374     int t;
375     cin >> t;
376
377     while (t--)
378     {
379         long long n, m, k, x, y;
380         int v = 0;
381
382         cin >> n >> m >> k;
383
384         for (x = 0; x*x <= k && x < n; x++)
385         {
386             if ((k-m*x)%(n-x) == 0)
387             {
388                 y = (k-m*x)/(n-x);
389                 if (y >= 0 && y < m && m*x + n*y - x*y == k)
390                 {
391                     v = 1;
392                 }
393             }
394         }
395
396         for (y = 0; y*y <= k && y < m; y++)
397         {
398             if ((k-n*y)%(m-y) == 0)
399             {
400                 x = (k-n*y)/(m-y);
401                 if (x >= 0 && x < n && m*x + n*y - x*y == k)
402                 {
403                     v = 1;
404                 }
405             }
406         }
407
408         if (n*m == k)
409         {
410             v = 1;
411         }
412
413         if (v == 1)
414         {

```

```

415         cout << "Yes\n";
416     }
417     else
418     {
419         cout << "No\n";
420     }
421 }
422 }
423
424 //MAKEPAL3
425
426 #ifdef WTSH
427     #include <wtsh.h>
428 #else
429     #include <bits/stdc++.h>
430     using namespace std;
431     #define dbg(...)
432 #endif
433
434 #define int long long
435 #define endl "\n"
436 #define sz(w) (int)(w.size())
437 using pii = pair<int, int>;
438
439 // ----- Input Checker Start -----
440
441 long long readInt(long long l, long long r, char endd)
442 {
443     long long x = 0;
444     int cnt = 0, fi = -1;
445     bool is_neg = false;
446     while(true)
447     {
448         char g = getchar();
449         if(g == '-')
450         {
451             assert(fi == -1);
452             is_neg = true;
453             continue;
454         }
455         if('0' <= g && g <= '9')
456         {
457             x *= 10;
458             x += g - '0';
459             if(cnt == 0)
460                 fi = g - '0';
461             cnt++;
462             assert(fi != 0 || cnt == 1);
463             assert(fi != 0 || is_neg == false);
464             assert(!(cnt > 19 || (cnt == 19 && fi > 1)));
465         }
466         else if(g == endd)
467         {
468             if(is_neg)
469                 x = -x;
470             if(!(l <= x && x <= r))
471             {
472                 cerr << "L: " << l << ", R: " << r << ", Value Found: " << x << '\n';
473                 assert(false);
474             }
475             return x;
476         }
477         else
478         {
479             assert(false);
480         }
481     }
482 }
483

```

```

484 string readString(int l, int r, char endd)
485 {
486     string ret = "";
487     int cnt = 0;
488     while(true)
489     {
490         char g = getchar();
491         assert(g != -1);
492         if(g == endd)
493             break;
494         cnt++;
495         ret += g;
496     }
497     assert(l <= cnt && cnt <= r);
498     return ret;
499 }
500
501 long long readIntSp(long long l, long long r) { return readInt(l, r, ' '); }
502 long long readIntLn(long long l, long long r) { return readInt(l, r, '\n'); }
503 string readStringSp(int l, int r) { return readString(l, r, ' '); }
504 string readStringLn(int l, int r) { return readString(l, r, '\n'); }
505 void readEOF() { assert(getchar() == EOF); }
506
507 vector<int> readVectorInt(int n, long long l, long long r)
508 {
509     vector<int> a(n);
510     for(int i = 0; i < n - 1; i++)
511         a[i] = readIntSp(l, r);
512     a[n - 1] = readIntLn(l, r);
513     return a;
514 }
515
516 // ----- Input Checker End -----
517
518 int sumN = 0;
519
520 int sign(int x)
521 {
522     return x >= 0 ? 1 : -1;
523 }
524
525 void solve()
526 {
527     int n = readIntLn(1, 1e5);
528     sumN += n;
529     vector<int> a = readVectorInt(n, 1, 1e9);
530     vector<int> d(n / 2);
531     for(int i = 0; i < n / 2; i++)
532         d[i] = a[i] - a[n - 1 - i];
533     int ans = 0;
534     for(int l = 0; l < sz(d);)
535     {
536         int r = l;
537         while(r < sz(d) and sign(d[l]) == sign(d[r]))
538             r++;
539         ans += abs(d[l]);
540         for(int i = l + 1; i < r; i++)
541             ans += max(0LL, abs(d[i]) - abs(d[i - 1]));
542         l = r;
543     }
544     cout << ans << endl;
545 }
546
547 int32_t main()
548 {
549     ios::sync_with_stdio(0);
550     cin.tie(0);
551     int T = readIntLn(1, 1e5);
552     for(int tc = 1; tc <= T; tc++)

```



```

553     {
554         solve();
555     }
556     readEOF();
557     assert(sumN <= 3e5);
558     return 0;
559 }
560
561 //MNVST
562 #include <bits/stdc++.h>
563 using namespace std;
564
565 const int maxn = 5000 + 10;
566 int dp[maxn][maxn];
567
568 int main () {
569     ios_base::sync_with_stdio(false), cin.tie(0);
570
571     memset(dp, 63, sizeof(dp));
572     for (int i = 0; i < maxn; i++) {
573         dp[0][i] = 0;
574     }
575
576     for (int i = 1; i < maxn; i++) {
577         for (int j = 1; j < maxn; j++) {
578             dp[i][j] = dp[i][j-1];
579             if (j <= i) dp[i][j] = min(dp[i][j], dp[i-j][j-1] + j*j);
580         }
581     }
582
583     int t;
584     cin >> t;
585
586     while (t--) {
587         int n, m;
588         cin >> n >> m;
589         n = min(n, m);
590         cout << dp[n][n];
591         if (t) cout << "\n";
592     }
593
594     return 0;
595 }
596
597 //GOODBINSTR
598 #include <bits/stdc++.h>
599
600 using namespace std;
601
602 int main()
603 {
604     int t;
605     cin >> t;
606
607     while (t--)
608     {
609         string s;
610         cin >> s;
611
612         if (s[0] == s[s.size()-1])
613         {
614             cout << s.size()-2 << "\n";
615         }
616         else
617         {
618             cout << "2\n";
619         }
620     }
621 }

```

```

622
623 //BIN_OD
624 // Code by Sahil Tiwari (still_me)
625
626 #include<bits/stdc++.h>
627 #define still_me main
628 #define endl "\n"
629 #define int long long int
630 #define all(a) (a).begin() , (a).end()
631 #define print(a) for(auto TEMPORARY: a) cout<<TEMPORARY<<" ";cout<<endl;
632 #define tt int TESTCASE;cin>>TESTCASE;while(TESTCASE--)
633 #define arrin(a,n) for(int INPUT=0;INPUT<n;INPUT++)cin>>a[INPUT]
634
635 using namespace std;
636 const int mod = 1e9+7;
637 const int inf = 1e18;
638
639 void solve() {
640     int n , q;
641     cin>>n>>q;
642     vector<int> a(n);
643     arrin(a , n);
644     vector<vector<int>>> b(n+1 , vector<int>(61));
645     for(int i=0;i<n;i++) {
646         for(int j=0;j<61;j++) {
647             if(a[i] & (1ll << j))
648                 b[i+1][j]++;
649             b[i+1][j] += b[i][j];
650         }
651     }
652     while(q--) {
653         int k , l , r , x , y;
654         cin>>k>>l>>r>>x>>y;
655         int o1 = b[r][k] - b[l-1][k];
656         int o2 = b[y][k] - b[x-1][k];
657         int z1 = r-l+1 - o1;
658         int z2 = y-x+1 - o2;
659         cout<<(o1*z2 + o2*z1)<<endl;
660     }
661 }
662
663 signed still_me()
664 {
665     ios_base::sync_with_stdio(false);cin.tie(NULL);cout.tie(NULL);
666
667     tt{
668         solve();
669     }
670     return 0;
671 }
672
673
674 //RJSTRING
675 #include <iostream>
676 #include <assert.h>
677 #include <algorithm>
678 #include <vector>
679 #include <set>
680 #include <string>
681 #include <queue>
682 #include <map>
683 #include<bits/stdc++.h>
684 #define ll int
685 #define timetaken cerr<<fixed<<setprecision(10); cerr << "time taken : " <<
(float)clock() / CLOCKS_PER_SEC << " secs" << endl
686 #define pb push_back
687 using namespace std;
688
689

```

```

690 long long readInt(long long l,long long r,char endd){
691     long long x=0;
692     int cnt=0;
693     int fi=-1;
694     bool is_neg=false;
695     while(true){
696         char g=getchar();
697         if(g=='-'){
698             assert(fi==-1);
699             is_neg=true;
700             continue;
701         }
702         if('0'<=g && g<='9'){
703             x*=10;
704             x+=g-'0';
705             if(cnt==0){
706                 fi=g-'0';
707             }
708             cnt++;
709             assert(fi!=0 || cnt==1);
710             assert(fi!=0 || is_neg==false);
711
712             assert(!(cnt>19 || ( cnt==19 && fi>1) ));
713         } else if(g==endd){
714             if(is_neg){
715                 x= -x;
716             }
717             assert(l<=x && x<=r);
718             return x;
719         } else {
720             assert(false);
721         }
722     }
723 }
724 string readString(int l,int r,char endd){
725     string ret="";
726     int cnt=0;
727     while(true){
728         char g=getchar();
729         assert(g!=-1);
730         if(g==endd){
731             break;
732         }
733         cnt++;
734         ret+=g;
735     }
736     assert(l<=cnt && cnt<=r);
737     return ret;
738 }
739 long long readIntSp(long long l,long long r){
740     return readInt(l,r,' ');
741 }
742 long long readIntLn(long long l,long long r){
743     return readInt(l,r,'\n');
744 }
745 string readStringLn(int l,int r){
746     return readString(l,r,'\n');
747 }
748 string readStringSp(int l,int r){
749     return readString(l,r,' ');
750 }
751 #ifndef ONLINE_JUDGE
752 #define dbg(x) cerr << #x << " : "; _print_(x);cerr << endl;
753 #else
754 #define dbg(x)
755 #endif
756 void _print_(ll t) {cerr << t;}
757 // void _print_(int t) {cerr << t;}
758 void _print_(string t) {cerr << t;}

```

```

759 void _print_(char t) {cerr << t;}
760 // void _print_(ld t) {cerr << t;}
761 void _print_(double t) {cerr << t;}
762 template <class T, class V> void _print_(pair <T, V> p);
763 template <class T> void _print_(vector <T> v);
764 template <class T> void _print_(set <T> v);
765 template <class T, class V> void _print_(map <T, V> v);
766 template <class T> void _print_(multiset <T> v);
767 template <class T, class V> void _print_(pair <T, V> p) {cerr << "{"; _print_(p.first);
cerr << ","; _print_(p.second); cerr << "}";}
768 template <class T> void _print_(vector <T> v) {cerr << "["; for (T i : v) {_print_(i);
cerr << " ";} cerr << "];"}
769 template <class T> void _print_(set <T> v) {cerr << "["; for (T i : v) {_print_(i);
cerr << " ";} cerr << "];"}
770 template <class T> void _print_(multiset <T> v) {cerr << "["; for (T i : v)
_print_(i); cerr << " ";} cerr << "];"}
771 template <class T, class V> void _print_(map <T, V> v) {cerr << "["; for (auto i : v)
_print_(i); cerr << " ";} cerr << "];"}
772 unordered_map<ll, string> m;
773 bool pref(string &s1, string &s2) // Determines if S1 is a prefix of S2
774 {
775     if(s1.length()>s2.length())
776         return 0;
777     for(ll i=0; i<s1.length(); i++)
778     {
779         if(s1[i]!=s2[i])
780             return 0;
781     }
782     return 1;
783 }
784 bool canmakebigger(vector<ll> freq, string &s1, string &s2) // Determines if we can
append characters in S1 such that it is > S2 Lexicographically
785 {
786     ll curr = 25;
787     while(curr>=0 && (freq[curr]==0))
788         curr--;
789     for(ll i=s1.length(); i<s2.length() && curr>=0; i++)
790     {
791         if(s2[i]<char(curr+'a'))
792             return 1;
793         if(s2[i]>char(curr+'a'))
794             return 0;
795         freq[s2[i]-'a']--;
796         while(curr>=0 && (freq[curr]==0))
797             curr--;
798     }
799     if(curr<0)
800         return 0;
801     return 1;
802 }
803 bool canconstruct(string s, string &s1, string &s2, ll start, vector<ll> &freq) // Can
we create a string s (initially empty), such that we can make it >s1 and <=s2
804 {
805     map<char,ll> m1;
806     for(ll i=0; i<26; i++)
807     {
808         if(freq[i]>0)
809             m1[i+'a'] = freq[i];
810     }
811     bool check = 0; // If check = 1, we have established that s2>s1 and s2>s
irrespective of the characters we append now
812     for(ll i=start; i<s1.size() && m1.size()>0; i++)
813     {
814         if(check)
815         {
816             char ch = (*--m1.end()).first; // Last Character Required Only
817             if(ch>s1[i])
818                 return 1;
819             if(ch<s1[i])

```

```

820         return 0;
821     if(ch==s1[i])
822     {
823         s.pb(ch);
824         m1[ch]--;
825         if(m1[ch]==0)
826             m1.erase(ch);
827     }
828 }
829 else
830 {
831     if(s1[i]<s2[i])
832         check = 1;
833     char ch1 = '.', ch2 = '.';
834     auto itr = m1.upper_bound(s1[i]);
835     if(itr!=m1.end())
836         ch2 = (*itr).first;
837     if(itr!=m1.begin())
838     {
839         itr--;
840         ch1 = (*itr).first;
841     }
842     // Ch1 and Ch2 - We only require these 2 characters to check for the
843     condition s>s1 && s<=s2
844     if((ch1!='.' && ch1>s1[i] && ch1<=s2[i]) || (ch2!='.' && ch2>s1[i] &&
845     ch2<=s2[i]))
846         return 1;
847     if((ch1!='.' && ch1==s1[i]) || (ch2!='.' && ch2==s1[i]))
848     {
849         char ch = s1[i];
850         s.pb(ch);
851         m1[ch]--;
852         if(m1[ch]==0)
853             m1.erase(ch);
854         continue;
855     }
856     return 0;
857 }
858 if(s==s1 && m1.size()>0)
859     s.pb((*m1.begin()).first);
860 return (s>s1 && s<=s2);
861 }
862 void solve(ll &n, string s, vector<ll> freq, ll &k, vector<string> &vect, ll &x)
863 {
864     if(m.find(x)!=m.end())
865         return;
866     if(x==0) // Base Case
867     {
868         if(s <= vect[0])
869             m[x] = "Yes";
870         else
871             m[x] = "No";
872     }
873     else if(x==k) // Base Case
874     {
875         if(s > vect.back())
876             m[x] = "Yes";
877         else if(pref(s,vect.back()) && canmakebigger(freq,s,vect.back())) // 4,5,6
878             m[x] = "Yes";
879         else
880             m[x] = "No";
881     }
882     else
883     {
884         string prev = vect[x-1], next = vect[x];
885         if((prev==next) || (s>next)) // Base Case
886         {
887             m[x] = "No";
888         }
889     }
890 }

```

```

887         return;
888     }
889     if(s > prev)
890         m[x] = "Yes";
891     else if(s < prev)
892     {
893         if(pref(s,prev) && pref(s,next))
894         {
895             if(canconstruct(s,prev,next,s.size(),freq))
896                 m[x] = "Yes";
897             else
898                 m[x] = "No";
899         }
900     else if(pref(s,prev))
901     {
902         if(canmakebigger(freq,s,prev))
903             m[x] = "Yes";
904         else
905             m[x] = "No";
906     }
907     else
908         m[x] = "No";
909 }
910 else
911 {
912     for(ll i=0; i<26; i++)
913     {
914         if(freq[i]>0)
915         {
916             s.pb(char(i+'a'));
917             break;
918         }
919     }
920     if(s>prev && s<=next)
921         m[x] = "Yes";
922     else
923         m[x] = "No";
924 }
925 }
926 }
927 int main()
928 {
929     #ifndef ONLINE_JUDGE
930         freopen("15.in", "r", stdin);
931         freopen("15.out", "w", stdout);
932     #endif
933     int N=readIntLn(1,1e4); // Ensures that an integer in the range [1, 3] is inputted,
    and that there is a New Line (Ln) right after that. This needs Unix-style line
    endings (ie. \n instead of \r\n). So generate the test files on an Unix machine
    (eg. Linux, Mac).
934     string s = readStringLn(N, N);
935     vector<ll> freq(26);
936     for(ll i=0; i<26; i++)
937     {
938         if(i==25)
939             freq[i] = readIntLn(0, 1e6);
940         else
941             freq[i] = readIntSp(0, 1e6);
942     }
943     int K = readIntLn(1, 1e4);
944     vector<string> vect(K);
945     int sum1 = 0;
946     for(int i=0; i<K; i++)
947     {
948         vect[i] = readStringLn(1, 1e4);
949         sum1+=vect[i].size();
950     }
951     assert(sum1<=1e5);
952     sort(vect.begin(), vect.end());

```

```

953     for(int i=0; i<=K; i++)
954     {
955         solve(N, s, freq, K, vect, i);
956         cout<<m[i]<<endl;
957     }
958     assert(getchar()==-1); // Ensures that there are no extra characters at the end.
959     cerr<<"SUCCESS\n"; // You should see this on the
    http://campus.codechef.com/files/stderr/SUBMISSION\_ID page, at the bottom.
960     timetaken;
961 }
962
963 //COPOINTS
964 #include<bits/stdc++.h>
965 #define int long long int
966 #define endl "\n"
967 #define quick ios_base::sync_with_stdio(false);cin.tie(NULL);cout.tie(NULL)
968
969 using namespace std;
970
971 signed main()
972 {
973     quick;
974
975     int t;
976     cin>>t;
977     while(t--){
978         int n;
979         cin>>n;
980         int a = 0, b = 0, c = 0, d = 0, o = 0;
981         while(n--){
982             int x , y;
983             cin>>x>>y;
984             if(x==0 && y==0){
985                 o++;
986             }
987             else if(y>0)
988                 c++;
989             else if(y<0)
990                 d++;
991             else if(x>0)
992                 a++;
993             else
994                 b++;
995         }
996
997         // cout<<a<<" "<<b<<" "<<c<<" "<<d<<endl;
998
999         int k = a+b+o;
1000         int A = k%2 ? k/2*(k/2+1) : k*k / 4;
1001
1002         k = c+d+o;
1003         int B = k%2 ? k/2*(k/2+1) : k*k / 4;
1004
1005         int X = A - a*b;
1006         int Y = B - c*d;
1007
1008         // cout<<A<<" "<<B<<endl;
1009
1010         int ans = A + B - min(a*b , Y) - min(c*d , X);
1011         X -= min(c*d , X);
1012         Y -= min(a*b , Y);
1013
1014         ans -= min(X , Y);
1015
1016         ans = max(ans , max({min(a , c) , min(a , d) , min(b , d) , min(b , c)}));
1017
1018         cout<<ans<<endl;
1019     }
1020

```

```

1021     cerr << "time taken : " << (float)clock() / CLOCKS_PER_SEC << " secs" << endl;
1022     return 0;
1023 }
1024
1025 //WGRAPH
1026 // Code by Sahil Tiwari (still_me)
1027
1028 #include<bits/stdc++.h>
1029 #define still_me main
1030 #define tt int TESTCASE;cin>>TESTCASE;while(TESTCASE--)
1031
1032 using namespace std;
1033
1034 void chal() {
1035     int n;
1036     cin>>n;
1037     cout<<min(4 , (n+1)/2)<<endl;
1038     vector<int> a(n+1);
1039     if(n < 8) {
1040         // cout<<(n+1)/2<<endl;
1041         for(int i=1;i<=n;i++) {
1042             a[i] = (i+1)/2;
1043         }
1044         if(n == 5) {
1045             swap(a[3] , a[5]);
1046             a[4] = a[3];
1047         }
1048     }
1049     else {
1050         // cout<<4<<endl;
1051         if(n % 4 == 0 || n % 4 == 3) {
1052             for(int i=1;i<=n;i++) {
1053                 a[i] = (i % 4);
1054                 if(!a[i])
1055                     a[i] = 4;
1056             }
1057         }
1058         else {
1059             for(int i=1;i<=n;i++) {
1060                 if(i % 4 == 1) {
1061                     a[i] = 4;
1062                 }
1063                 else if(i % 4) {
1064                     a[i] = i % 4;
1065                 }
1066                 else {
1067                     a[i] = 1;
1068                 }
1069             }
1070         }
1071     }
1072
1073     set<pair<int,int>> b;
1074     for(int i=1;i<=n;i++) {
1075         b.insert({a[i] , i});
1076     }
1077     set<pair<int,int>> edges;
1078     while(!b.empty()) {
1079         pair<int,int> x = *b.begin();
1080         b.erase(x);
1081         pair<int,int> y;
1082         vector<pair<int,int>> c;
1083         while(x.first) {
1084             x.first--;
1085             y = *b.rbegin();
1086             // cout<<b.size()<<endl;
1087             b.erase(y);
1088             c.push_back(y);

```



```

1090         edges.insert({x.second , y.second});
1091         // cout<<x.second<<" "<<y.second<<endl;
1092     }
1093     for(auto i: c) {
1094         b.insert({i.first-1 , i.second});
1095     }
1096 }
1097 cout<<edges.size()<<"\n";
1098 for(auto i: edges) {
1099     cout<<i.first<<" "<<i.second<<"\n";
1100 }
1101 // cout<<"done\n";
1102 }
1103
1104 signed still_me()
1105 {
1106     ios_base::sync_with_stdio(false);cin.tie(NULL);cout.tie(NULL);
1107
1108     tt{
1109         chal();
1110     }
1111     // cout<<"done\n";
1112     return 0;
1113 }
1114
1115 //COLORS
1116 // Code by Sahil Tiwari (still_me)
1117
1118 #include <bits/stdc++.h>
1119 #define still_me main
1120 #define print(a) for(auto TEMPORARY: a) cout<<TEMPORARY<<" ";
1121 #define tt int TESTCASE;cin>>TESTCASE;while(TESTCASE--)
1122
1123 using namespace std;
1124 const int mod = 1e9+7;
1125 const int inf = 1e18;
1126
1127 void solve() {
1128     int n , k;
1129     cin>>n>>k;
1130     int m = n/k;
1131     int ans = (n+m-1) / m;
1132     cout<<ans<<endl;
1133     vector<vector<int>>> a(m);
1134     int c = 1;
1135     while(n) {
1136         for(int i=0;i<m && n;i++) {
1137             a[i].push_back(c);
1138             n--;
1139         }
1140         c++;
1141     }
1142     for(auto i: a){
1143         print(i);
1144     }
1145 }
1146
1147 signed still_me()
1148 {
1149     ios_base::sync_with_stdio(false);cin.tie(NULL);cout.tie(NULL);
1150
1151     tt{
1152         solve();
1153     }
1154     return 0;
1155 }
1156
1157 //SIMPLEARRY
1158 #include <bits/stdc++.h>

```

```

1159 #define int long long int
1160 using namespace std;
1161
1162 #define mod 1000000007
1163
1164 int power(int a , int b) {
1165     if(b == 0)
1166         return 1;
1167     int res = power(a , b>>1);
1168     if(b & 1)
1169         return (res * res % mod) * a % mod;
1170     return res * res % mod;
1171 }
1172
1173 signed main() {
1174
1175     int t;
1176     cin>>t;
1177     while(t--) {
1178         int n , k;
1179         cin>>n>>k;
1180         vector<int> a(k);
1181         for(int i=0;i<n;i++) {
1182             int x;
1183             cin>>x;
1184             a[x % k]++;
1185         }
1186         // for(auto &i: a)
1187         //     cout<<i<<" ";
1188         // cout<<endl;
1189         int ans = 1;
1190         // cout<<power(2 , 2)<<endl;
1191         for(int i=1;i<(k+1)/2;i++) {
1192             int c = (power(2 , a[i]) + power(2 , a[k-i]) - 1);
1193             ans = ans * c % mod;
1194         }
1195         if(k % 2 == 0) {
1196             ans = ans * (a[k/2] + 1) % mod;
1197         }
1198         ans = ans * (a[0]+1) % mod;
1199         cout<<ans<<endl;
1200     }
1201     return 0;
1202 }
1203
1204 //LAZYANC
1205 #include<bits/stdc++.h>
1206 #define ll long long int
1207 #define fab(a,b,i) for(int i=a;i<b;i++)
1208 #define pb push_back
1209 #define db double
1210 #define mp make_pair
1211 #define endl "\n"
1212 #define f first
1213 #define se second
1214 #define all(x) x.begin(),x.end()
1215 #define vll vector<ll>
1216 #define vi vector<int>
1217 #define pii pair<int,int>
1218 #define pll pair<ll,ll>
1219 #define quick ios_base::sync_with_stdio(false);cin.tie(NULL);cout.tie(NULL)
1220
1221 using namespace std;
1222
1223 const int MOD = 1e9 + 7;
1224
1225 ll add(ll x, ll y) {ll res = x + y; return (res >= MOD ? res - MOD : res);}
1226 ll mul(ll x, ll y) {ll res = x * y; return (res >= MOD ? res % MOD : res);}
1227 ll sub(ll x, ll y) {ll res = x - y; return (res < 0 ? res + MOD : res);}

```

```

1228 ll power(ll x, ll y) {ll res = 1; x %= MOD; while (y) {if (y & 1)res = mul(res, x); y
1229 >>= 1; x = mul(x, x);} return res;}
1230 ll mod_inv(ll x) {return power(x, MOD - 2);}
1231 ll lcm(ll x, ll y) { ll res = x / __gcd(x, y); return (res * y);}
1232
1233 #define int ll
1234 void dfs(int src, int par, vector<int> &a, vector<vector<int>> &v, vector<vector<int>>
1235 &values, vector<int> &parent) {
1236     parent[src] = par;
1237     for (int &i : v[src]) {
1238         if (i ^ par) {
1239             dfs(i, src, a, v, values, parent);
1240         }
1241     }
1242
1243     int curr = src;
1244     int val = a[src];
1245     while (curr != -1 and val > 0) {
1246         values[curr].push_back(val);
1247         curr = parent[curr];
1248         val >>= 1;
1249     }
1250 }
1251
1252 int32_t main()
1253 {
1254
1255     quick;
1256     int t = 1;
1257     cin >> t;
1258     while (t--)
1259     {
1260         int n;
1261         cin >> n;
1262         vector<vector<int>> v(n);
1263
1264         for (int i = 0; i < n - 1; i++) {
1265             int x, y;
1266             cin >> x >> y;
1267             x--, y--;
1268             v[x].push_back(y);
1269             v[y].push_back(x);
1270         }
1271
1272         vector<int> a(n);
1273         for (int i = 0; i < n; i++) {
1274             cin >> a[i];
1275         }
1276
1277         vector<vector<int>> values(n);
1278         vector<int> parent(n, -1);
1279         dfs(0, -1, a, v, values, parent);
1280
1281         vector<int> ans(n);
1282
1283         for (int &i : values[0]) {
1284             ans[0] += i;
1285         }
1286
1287         const int maxA = 1e6 + 5;
1288
1289         const int N = log2(maxA) + 3;
1290
1291         vector<vector<int>> moveNodes(n, vector<int> (N));
1292
1293         for (int i = 0; i < n; i++) {
1294             for (int &j : values[i]) {

```

```

1295         for (int k = 0; k < N; k++) {
1296             int val = (j >> k);
1297             moveNodes[i][k] += val;
1298             if (val == 0) break;
1299         }
1300     }
1301 }
1302
1303 for (int i = 1; i < n; i++) {
1304
1305     ans[i] = moveNodes[i][0];
1306     int last = i;
1307     int curr = parent[i];
1308     for (int j = 1; j < N - 1 and curr != -1; j++) {
1309         int val = (moveNodes[curr][j] - moveNodes[last][j + 1]);
1310         ans[i] += val;
1311         last = curr;
1312         curr = parent[curr];
1313     }
1314 }
1315
1316
1317 for (int i = 0; i < n; i++) cout << ans[i] << " ";
1318 cout << endl;
1319
1320
1321 }
1322
1323
1324
1325     cerr << "time taken : " << (float)clock() / CLOCKS_PER_SEC << " secs" << endl;
1326     return 0;
1327 }
1328
1329 //TREEQUERIES
1330 #include <bits/stdc++.h>
1331 using namespace std;
1332 #define int int64_t
1333 #define sp << ' ' <<
1334 #define nl << '\n'
1335
1336 const int Z = 3e5, B = 20;
1337
1338 // INPUT, 0-indexed
1339 int N;
1340 vector<int> g[Z];
1341 // .....
1342
1343 int dfsTimer, tin[Z], sz[Z], t[Z], e[Z], p[Z][B], q[Z][B], l[Z][B], r[Z][B];
1344 bool h[Z][B];
1345
1346 void dfs(int u) {
1347     e[tin[u] = dfsTimer++] = u;
1348
1349     for(int i = 0; i + 1 < B; ++i) {
1350         p[u][i+1] = p[p[u][i]][i];
1351         q[u][i+1] = q[p[u][i]][i];
1352         h[u][i+1] = h[p[u][i]][i];
1353     }
1354     sort(begin(g[u]), end(g[u]));
1355
1356     for(int v : g[u]) {
1357         if(v != p[u][0]) {
1358             p[v][0] = u;
1359             q[v][0] = v;
1360             h[v][0] = 1;
1361             dfs(v);
1362         } else
1363             l[u][0] = dfsTimer - tin[u];

```

```

1364     }
1365     sz[u] = dfsTimer - tin[u];
1366
1367     int add {};
1368     for(int v : g[u]) {
1369         if(v != p[u][0]) t[v] = tin[v] - tin[u] + add;
1370         else add = N - sz[u];
1371     }
1372
1373     r[u][0] = l[u][0] + (N - sz[u]) - 1;
1374 }
1375
1376 int query(int u, int x) {
1377     for(int i = B; i--; ) if(h[u][i]) {
1378         if(l[u][i] <= x && x <= r[u][i]) {
1379             x -= l[u][i];
1380             if(t[q[u][i]] <= x) x += sz[q[u][i]];
1381             u = p[u][i];
1382         }
1383     }
1384     if(!u) return e[x];
1385
1386     if(x < l[u][0]) return e[tin[u] + x];
1387     return e[tin[u] + x - (N - sz[u])];
1388 }
1389
1390 signed main() {
1391     cin.tie(0)->sync_with_stdio(0);
1392
1393     int T; cin >> T;
1394     while(T--) {
1395         cin >> N;
1396         for(int i = 0; i < N; ++i) {
1397             g[i].clear();
1398             fill(h[i], h[i] + B, 0);
1399         }
1400         dfsTimer = 0;
1401
1402         for(int i = 1; i < N; ++i) {
1403             int u, v; cin >> u >> v;
1404             --u, --v;
1405             g[u].push_back(v);
1406             g[v].push_back(u);
1407         }
1408
1409         dfs(0);
1410
1411         for(int i = 0; i + 1 < B; ++i) {
1412             for(int u = 1; u < N; ++u) {
1413                 int &lv = l[u][i+1] = l[p[u][i]][i];
1414                 int &rv = r[u][i+1] = r[p[u][i]][i];
1415
1416                 if(p[u][i] && q[u][i] < p[p[u][i]][0]) {
1417                     lv -= sz[q[u][i]];
1418                     rv -= sz[q[u][i]];
1419                 }
1420
1421                 lv += l[u][i];
1422                 rv += l[u][i];
1423             }
1424         }
1425
1426         int Q, last {}; cin >> Q;
1427         while(Q--) {
1428             int u, x; cin >> u >> x;
1429             cout << (last = query((u ^ last) - 1, x ^ last) + 1) nl;
1430         }
1431     }
1432 }

```

```

1433
1434 //PRIMEFACDIV
1435 #include "bits/stdc++.h"
1436 using namespace std;
1437 /*
1438 #include <ext/pb_ds/assoc_container.hpp>
1439 using namespace __gnu_pbds;
1440 using ordered_set = tree<int, null_type, less<int>, rb_tree_tag,
1441 tree_order_statistics_node_update>;
1442 */
1443 #define all(x) begin(x), end(x)
1444 #define rall(x) rbegin(x), rend(x)
1445 #define sz(x) (int)(x).size()
1446
1447 using ll = long long;
1448 #define int ll
1449 const int mod = 1e9+7;
1450
1451 void solve(int tc) {
1452     int a, b;
1453     cin >> a >> b;
1454     int g = gcd(a, b);
1455     int gg = gcd(g, b);
1456     while (b % gg == 0 and gg != 1) {
1457         b /= gg;
1458         gg = gcd(gg, b);
1459     }
1460     cout << (b == 1 ? "YES\n" : "NO\n");
1461 }
1462
1463 signed main() {
1464     cin.tie(0)->sync_with_stdio(0);
1465     int tc = 1;
1466     cin >> tc;
1467     for (int i = 1; i <= tc; ++i) solve(i);
1468     return 0;
1469 }
1470
1471 //DISTINCTNEIG
1472 #pragma GCC optimize("O3")
1473 #pragma GCC optimize("Ofast,unroll-loops")
1474 #include <bits/stdc++.h>
1475 #include <ext/pb_ds/tree_policy.hpp>
1476 #include <ext/pb_ds/assoc_container.hpp>
1477 using namespace __gnu_pbds;
1478 using namespace std;
1479 #define ll long long
1480 const ll INF_MUL=1e13;
1481 const ll INF_ADD=1e18;
1482 #define pb push_back
1483 #define mp make_pair
1484 #define nline "\n"
1485 #define f first
1486 #define s second
1487 #define pll pair<ll,ll>
1488 #define all(x) x.begin(),x.end()
1489 #define vl vector<ll>
1490 #define vvl vector<vector<ll>>
1491 #define vvvl vector<vector<vector<ll>>>
1492 #ifndef ONLINE_JUDGE
1493 #define debug(x) cerr<<#x<<" "; _print(x); cerr<<nline;
1494 #else
1495 #define debug(x);
1496 #endif
1497 void _print(ll x){cerr<<x;}
1498 void _print(char x){cerr<<x;}
1499 void _print(string x){cerr<<x;}
1500 mt1937 rng(chrono::steady_clock::now().time_since_epoch().count());

```

```

1501 template<class T,class V> void _print(pair<T,V> p) {cerr<<"{";
1502 _print(p.first);cerr<<" "; _print(p.second);cerr<<"}";}
1503 template<class T>void _print(vector<T> v) {cerr<<" [ "; for (T i:v){_print(i);cerr<<"
1504 ";}cerr<<" ]";}
1505 template<class T>void _print(set<T> v) {cerr<<" [ "; for (T i:v){_print(i); cerr<<"
1506 ";}cerr<<" ]";}
1507 template<class T>void _print(multiset<T> v) {cerr<<" [ "; for (T
1508 i:v){_print(i);cerr<<" ";}cerr<<" ]";}
1509 template<class T,class V>void _print(map<T, V> v) {cerr<<" [ "; for(auto i:v)
1510 {_print(i);cerr<<" ";} cerr<<" ]";}
1511 typedef tree<ll, null_type, less<ll>, rb_tree_tag, tree_order_statistics_node_update>
1512 ordered_set;
1513 typedef tree<ll, null_type, less_equal<ll>, rb_tree_tag,
1514 tree_order_statistics_node_update> ordered_multiset;
1515 typedef tree<pair<ll,ll>, null_type, less<pair<ll,ll>>, rb_tree_tag,
1516 tree_order_statistics_node_update> ordered_pset;
1517 //-----
1518 -----
1519
1520 const ll MOD=1e9+7;
1521 const ll MAX=1000100;
1522 void solve(){
1523     ll n; cin>>n;
1524     vector<ll> freq(n+5,0);
1525     ll till=n*(n+1)/2;
1526     for(ll i=1;i<=2*n;i++){
1527         ll x; cin>>x;
1528         freq[x]++;
1529     }
1530     sort(all(freq));
1531     if(freq.back()>till){
1532         cout<<"NO\n";
1533     }
1534     else{
1535         cout<<"YES\n";
1536     }
1537     return;
1538 }
1539 int main()
1540 {
1541     ios_base::sync_with_stdio(false);
1542     cin.tie(NULL);
1543     #ifndef ONLINE_JUDGE
1544     freopen("input.txt", "r", stdin);
1545     freopen("output.txt", "w", stdout);
1546     freopen("error.txt", "w", stderr);
1547     #endif
1548     ll test_cases=1;
1549     cin>>test_cases;
1550     while(test_cases--){
1551         solve();
1552     }
1553     cout<<fixed<<setprecision(10);
1554     cerr<<"Time:"<<(1000*((double)clock())/(double)CLOCKS_PER_SEC)<<"ms\n";
1555 }
1556
1557 //SQUARESORT
1558 #include <bits/stdc++.h>
1559 using namespace std;
1560 #define int int64_t
1561 #define sp << ' ' <<
1562 #define nl << '\n'
1563
1564 int root(int x) {
1565     int y = sqrtl(x) + 2;
1566     while(y * y > x) --y;
1567     return y;
1568 }
1569

```

```

1560 signed main() {
1561     cin.tie(0)->sync_with_stdio(0);
1562
1563     int T; cin >> T;
1564     while(T--) {
1565         int N; cin >> N;
1566
1567         map<int, int> dp {{0, 0}}, cur;
1568
1569         while(N--) {
1570             int a; cin >> a;
1571
1572             cur[a] = 0;
1573             queue<int> q;
1574             q.push(a);
1575
1576             while(!empty(q)) {
1577                 int u = q.front(); q.pop();
1578
1579                 vector<int> vs;
1580                 if(1 < u) vs.push_back(root(u));
1581                 if(u <= (int)1e9) vs.push_back(u * u);
1582
1583                 for(int v : vs)
1584                     if(cur.find(v) == end(cur))
1585                         cur[v] = cur[u] + 1, q.push(v);
1586             }
1587
1588             int prefMin = 1e18;
1589
1590             for(auto &[i, j] : cur) {
1591                 auto k = dp.upper_bound(i);
1592                 j += prev(k)->second;
1593
1594                 j = min(j, prefMin);
1595                 prefMin = min(prefMin, j);
1596             }
1597
1598             swap(cur, dp);
1599             cur.clear();
1600         }
1601
1602         int ans = 1e18;
1603         for(auto [i, j] : dp)
1604             ans = min(ans, j);
1605
1606         cout << ans nl;
1607     }
1608 }
1609
1610 //RANGEASSIGN
1611 #include <bits/stdc++.h>
1612 using namespace std;
1613 #define int int64_t
1614 #define sp << ' ' <<
1615 #define nl << '\n'
1616
1617 signed main() {
1618     cin.tie(0)->sync_with_stdio(0);
1619
1620     int T; cin >> T;
1621     while(T--) {
1622         int N; cin >> N;
1623
1624         int A[N];
1625         for(int &i : A) cin >> i;
1626
1627         bool ok = A[0] == A[N-1];
1628

```



```

1629         for(int i = 1; i < N; ++i)
1630             if(A[i] == A[N-1] && A[i-1] == A[0]) ok = 1;
1631
1632         cout << (ok ? "YES" : "NO") nl;
1633     }
1634 }
1635
1636 //XORARRAY
1637 #include <bits/stdc++.h>
1638 using namespace std;
1639 #define int int64_t
1640 #define sp << ' ' <<
1641 #define nl << '\n'
1642
1643 signed main() {
1644     cin.tie(0)->sync_with_stdio(0);
1645
1646     int T; cin >> T;
1647     while(T--) {
1648         int N; cin >> N;
1649
1650         int A[N];
1651         for(int &i : A) cin >> i;
1652
1653         for(int i = N; --i; )
1654             A[i] = (A[i] < A[i-1] ? -1 : 1) * (64 - __builtin_clzll(A[i] ^ A[i-1]));
1655
1656         int ans = 1, _cnt[61] {}, *cnt = _cnt + 30, j = 1;
1657
1658         for(int i = 1; i < N; --cnt[A[i++]]) {
1659             for(; j < N && !cnt[-A[j]]; ++cnt[A[j++]]);
1660
1661             ans = max(ans, j - i + 1);
1662         }
1663
1664         cout << ans nl;
1665     }
1666 }
1667
1668 //COUNTPART
1669 #pragma GCC optimize("O3")
1670 #pragma GCC target("popcnt")
1671 #pragma GCC target("avx,avx2,fma")
1672 #pragma GCC optimize("Ofast,unroll-loops")
1673 #include <bits/stdc++.h>
1674 #include <ext/pb_ds/tree_policy.hpp>
1675 #include <ext/pb_ds/assoc_container.hpp>
1676 using namespace __gnu_pbds;
1677 using namespace std;
1678 #define ll long long
1679 const ll INF_MUL=1e13;
1680 const ll INF_ADD=1e18;
1681 #define pb push_back
1682 #define mp make_pair
1683 #define nline "\n"
1684 #define f first
1685 #define s second
1686 #define pll pair<ll,ll>
1687 #define all(x) x.begin(),x.end()
1688 #define vl vector<ll>
1689 #define vvl vector<vector<ll>>
1690 #define vvvl vector<vector<vector<ll>>>
1691 #ifndef ONLINE_JUDGE
1692 #define debug(x) cerr<<#x<<" "; _print(x); cerr<<nline;
1693 #else
1694 #define debug(x);
1695 #endif
1696 void _print(ll x){cerr<<x;}
1697 void _print(int x){cerr<<x;}

```

```

1698 void _print(char x){cerr<<x;}
1699 void _print(string x){cerr<<x;}
1700 mt19937 rng(chrono::steady_clock::now().time_since_epoch().count());
1701 template<class T,class V> void _print(pair<T,V> p) {cerr<<"{";
    _print(p.first);cerr<<","; _print(p.second);cerr<<"}";}
1702 template<class T>void _print(vector<T> v) {cerr<<" ["; for (T i:v){_print(i);cerr<<"
    ";}cerr<<"}";}
1703 template<class T>void _print(set<T> v) {cerr<<" ["; for (T i:v){_print(i); cerr<<"
    ";}cerr<<"}";}
1704 template<class T>void _print(multiset<T> v) {cerr<<" ["; for (T
    i:v){_print(i);cerr<<" ";}cerr<<"}";}
1705 template<class T,class V>void _print(map<T, V> v) {cerr<<" ["; for(auto i:v)
    {_print(i);cerr<<" ";} cerr<<"}";}
1706 typedef tree<ll, null_type, less<ll>, rb_tree_tag, tree_order_statistics_node_update>
    ordered_set;
1707 typedef tree<ll, null_type, less_equal<ll>, rb_tree_tag,
    tree_order_statistics_node_update> ordered_multiset;
1708 typedef tree<pair<ll,ll>, null_type, less<pair<ll,ll>>, rb_tree_tag,
    tree_order_statistics_node_update> ordered_pset;
1709 //-----
    -----
1710 const ll MOD=998244353;
1711 const ll MAX=100100;
1712 void solve(){
1713     ll n; cin>>n;
1714     vector<ll> dp(n+5,1);
1715     vector<ll> pref(n+5,1);
1716     vector<ll> a(n+5,0);
1717     vector<ll> track;
1718     ll till=1;
1719     for(ll i=1;i<=n;i++){
1720         cin>>a[i];
1721         while(!track.empty()){
1722             auto it=track.back();
1723             if(a[i]>a[it]){
1724                 track.pop_back();
1725             }
1726             else{
1727                 break;
1728             }
1729         }
1730         if(a[i]<a[i-1]){
1731             till=i;
1732         }
1733         dp[i]=pref[i-1];
1734         if(till!=1){
1735             dp[i]-=pref[till-2];
1736         }
1737         if(!track.empty()){
1738             dp[i]+=dp[track.back()];
1739         }
1740         track.push_back(i);
1741         dp[i]%=MOD;
1742         dp[i]=(dp[i]+MOD)%MOD;
1743         pref[i]=(pref[i-1]+dp[i])%MOD;
1744     }
1745     cout<<dp[n]<<endl;
1746     return;
1747 }
1748 int main()
1749 {
1750     ios_base::sync_with_stdio(false);
1751     cin.tie(NULL);
1752     #ifndef ONLINE_JUDGE
1753     freopen("input.txt", "r", stdin);
1754     freopen("output.txt", "w", stdout);
1755     freopen("error.txt", "w", stderr);
1756     #endif

```

```

1757     ll test_cases=1;
1758     cin>>test_cases;
1759     while(test_cases--){
1760         solve();
1761     }
1762     cout<<fixed<<setprecision(10);
1763     cerr<<"Time:"<<1000*((double)clock())/((double)CLOCKS_PER_SEC<<"ms\n";
1764 }
1765
1766 //GREEDGRID
1767 #include "bits/stdc++.h"
1768 // #pragma GCC optimize("O3,unroll-loops")
1769 // #pragma GCC target("avx2,bmi,bmi2,lzcnt,popcnt")
1770 using namespace std;
1771 using ll = long long int;
1772 mt19937_64 rng(chrono::high_resolution_clock::now().time_since_epoch().count());
1773
1774 int main()
1775 {
1776     ios::sync_with_stdio(false); cin.tie(0);
1777
1778     int t; cin >> t;
1779     while (t--) {
1780         int n, m; cin >> n >> m;
1781         vector grid(n, vector(m, 0));
1782         for (auto &row : grid) for (auto &x : row) cin >> x;
1783         vector dp(n, vector(m, 0));
1784         for (int i = n-1; i >= 0; --i) {
1785             for (int j = m-1; j >= 0; --j) {
1786                 if (i == n-1 and j == m-1) dp[i][j] = grid[i][j];
1787                 else if (i == n-1) dp[i][j] = max(grid[i][j], dp[i][j+1]);
1788                 else if (j == m-1) dp[i][j] = max(grid[i][j], dp[i+1][j]);
1789                 else {
1790                     if ((i+j)%2 == 0) { // Becky
1791                         dp[i][j] = max(grid[i][j], min(dp[i+1][j], dp[i][j+1]));
1792                     }
1793                     else { // Anya
1794                         if (grid[i+1][j] > grid[i][j+1]) dp[i][j] = max(grid[i][j],
1795                             dp[i+1][j]);
1796                         else dp[i][j] = max(grid[i][j], dp[i][j+1]);
1797                     }
1798                 }
1799             }
1800         }
1801         cout << dp[0][0] << '\n';
1802     }
1803
1804 //GCDSORT
1805 #pragma GCC optimize("O3")
1806 #pragma GCC target("popcnt")
1807 #pragma GCC target("avx,avx2,fma")
1808 #pragma GCC optimize("Ofast,unroll-loops")
1809 #include <bits/stdc++.h>
1810 #include <ext/pb_ds/tree_policy.hpp>
1811 #include <ext/pb_ds/assoc_container.hpp>
1812 using namespace __gnu_pbds;
1813 using namespace std;
1814 #define ll long long
1815 const ll INF_MUL=1e13;
1816 const ll INF_ADD=1e18;
1817 #define pb push_back
1818 #define mp make_pair
1819 #define nline "\n"
1820 #define f first
1821 #define s second
1822 #define pll pair<ll,ll>
1823 #define all(x) x.begin(),x.end()
1824 #define vl vector<ll>

```

```

1825 #define vvl vector<vector<ll>>
1826 #define vvvl vector<vector<vector<ll>>>
1827 #ifndef ONLINE_JUDGE
1828 #define debug(x) cerr<<"x<<" "; _print(x); cerr<<endl;
1829 #else
1830 #define debug(x);
1831 #endif
1832 void _print(ll x){cerr<<x;}
1833 void _print(int x){cerr<<x;}
1834 void _print(char x){cerr<<x;}
1835 void _print(string x){cerr<<x;}
1836 mt19937 rng(chrono::steady_clock::now().time_since_epoch().count());
1837 template<class T,class V> void _print(pair<T,V> p) {cerr<<"{";
1838 _print(p.first);cerr<<","; _print(p.second);cerr<<"}";}
1839 template<class T>void _print(vector<T> v) {cerr<<" [ "; for (T i:v){_print(i);cerr<<"
1840 ";}cerr<<"]";}
1841 template<class T>void _print(set<T> v) {cerr<<" [ "; for (T i:v){_print(i); cerr<<"
1842 ";}cerr<<"]";}
1843 template<class T>void _print(multiset<T> v) {cerr<<" [ "; for (T
1844 i:v){_print(i);cerr<<" ";}cerr<<"]";}
1845 template<class T,class V>void _print(map<T, V> v) {cerr<<" [ "; for(auto i:v)
1846 {_print(i);cerr<<" ";} cerr<<"]";}
1847 typedef tree<ll, null_type, less<ll>, rb_tree_tag, tree_order_statistics_node_update>
1848 ordered_set;
1849 typedef tree<ll, null_type, less_equal<ll>, rb_tree_tag,
1850 tree_order_statistics_node_update> ordered_multiset;
1851 typedef tree<pair<ll,ll>, null_type, less<pair<ll,ll>>, rb_tree_tag,
1852 tree_order_statistics_node_update> ordered_pset;
1853 //-----
1854 -----
1855 const ll MOD=998244353;
1856 const ll MAX=200200;
1857 ll track[20][MAX];
1858 vector<ll> use(MAX,1);
1859 vector<ll> anot(MAX,0);
1860 ll getv(ll l,ll r){
1861     ll len=r-l+1;
1862     len=use[len];
1863     ll x=anot[len];
1864     return __gcd(track[x][l],track[x][r-len+1]);
1865 }
1866 void solve(){
1867     ll n; cin>>n;
1868     vector<ll> a(n+5);
1869     for(ll i=1;i<=n;i++){
1870         cin>>a[i];
1871     }
1872     for(ll i=0;i<20;i++){
1873         track[i][n+1]=0;
1874     }
1875     for(ll i=n;i>=1;i--){
1876         track[0][i]=a[i];
1877         for(ll j=1;j<20;j++){
1878             ll len=use[j];
1879             ll r=min(i+len,n+1);
1880             track[j][i]=__gcd(track[j-1][i],track[j-1][r]);
1881         }
1882     }
1883     ll l=n-1;
1884     for(ll i=n-1;i>=1;i--){
1885         l=min(l,i);
1886         a[i]=__gcd(a[i],getv(l,i));
1887         if(a[i]<=a[i+1]){
1888             continue;
1889         }
1890         ll found=0;
1891         while(l>=2){
1892             l--;

```

```

1884         ll comp=getv(1,i);
1885         comp=__gcd(comp,a[i]);
1886         if(comp<=a[i+1]){
1887             a[i]=__gcd(a[i],comp);
1888             found=1;
1889             break;
1890         }
1891     }
1892     if(found==0){
1893         cout<<"NO\n";
1894         return;
1895     }
1896 }
1897 cout<<"YES\n";
1898 return;
1899 }
1900 int main()
1901 {
1902     ios_base::sync_with_stdio(false);
1903     cin.tie(NULL);
1904     #ifndef ONLINE_JUDGE
1905     freopen("input.txt", "r", stdin);
1906     freopen("output.txt", "w", stdout);
1907     freopen("error.txt", "w", stderr);
1908     #endif
1909     ll test_cases=1;
1910     cin>>test_cases;
1911     ll cur=2,pos=1;
1912     while(cur<MAX){
1913         use[cur]=cur;
1914         anot[cur]=pos;
1915         cur*=2; pos++;
1916     }
1917     for(ll i=2;i<MAX;i++){
1918         use[i]=max(use[i],use[i-1]);
1919     }
1920     while(test_cases--){
1921         solve();
1922     }
1923     cout<<fixed<<setprecision(10);
1924     cerr<<"Time:"<<1000*((double)clock())/((double)CLOCKS_PER_SEC)<<"ms\n";
1925 }
1926
1927 //MINABS
1928 #include<bits/stdc++.h>
1929 using namespace std;
1930
1931 #define mod 1000000007
1932 typedef set<string> ss;
1933 typedef vector<int> vs;
1934 typedef map<int, char> msi;
1935 typedef pair<int, int> pa;
1936 typedef long long int ll;
1937
1938
1939 int main()
1940 {
1941     ios_base::sync_with_stdio(false);
1942     cin.tie(0);
1943
1944     int t;
1945     cin >> t;
1946     while (t--)
1947     {
1948         int n;
1949         cin >> n;
1950         string a, b;
1951         cin >> a >> b;
1952         int ans = 0;

```

```

1953         for (int i = 0; i < n; i++)
1954             ans += b[i] - a[i];
1955         ans = (ans % 26 + 26) % 26;
1956         cout << min(ans, abs(26 - ans)) << "\n";
1957     }
1958
1959     return 0;
1960 }
1961
1962 //DELXORONE
1963 #include "bits/stdc++.h"
1964 // #pragma GCC optimize("O3,unroll-loops")
1965 // #pragma GCC target("avx2,bmi,bmi2,lzcnt,popcnt")
1966 using namespace std;
1967 using ll = long long int;
1968 mt19937_64 rng(chrono::high_resolution_clock::now().time_since_epoch().count());
1969
1970 int main()
1971 {
1972     ios::sync_with_stdio(false); cin.tie(0);
1973
1974     int t; cin >> t;
1975     while (t--) {
1976         int n; cin >> n;
1977         vector<int> freq(n+10);
1978         for (int i = 0; i < n; ++i) {
1979             int x; cin >> x;
1980             ++freq[x];
1981         }
1982         int ans = n - 1;
1983         for (int i = 0; i <= n; ++i) {
1984             ans = min(ans, n - freq[i] - freq[i^1]);
1985         }
1986         cout << ans << '\n';
1987     }
1988 }
1989
1990 //DISTNEIGH
1991 // #pragma GCC optimize("O3")
1992 // #pragma GCC target("popcnt")
1993 // #pragma GCC target("avx,avx2,fma")
1994 // #pragma GCC optimize("Ofast,unroll-loops")
1995 #include <bits/stdc++.h>
1996 #include <ext/pb_ds/tree_policy.hpp>
1997 #include <ext/pb_ds/assoc_container.hpp>
1998 using namespace __gnu_pbds;
1999 using namespace std;
2000 #define ll long long
2001 const ll INF_MUL=1e13;
2002 const ll INF_ADD=1e18;
2003 #define pb push_back
2004 #define mp make_pair
2005 #define nline "\n"
2006 #define f first
2007 #define s second
2008 #define pll pair<ll,ll>
2009 #define all(x) x.begin(),x.end()
2010 #define vl vector<ll>
2011 #define vvl vector<vector<ll>>
2012 #define vvvl vector<vector<vector<ll>>>
2013 #ifndef ONLINE_JUDGE
2014 #define debug(x) cerr<<#x<<" "; _print(x); cerr<<nline;
2015 #else
2016 #define debug(x);
2017 #endif
2018 void _print(ll x){cerr<<x;}
2019 void _print(char x){cerr<<x;}
2020 void _print(string x){cerr<<x;}
2021 mt19937 rng(chrono::steady_clock::now().time_since_epoch().count());

```

```

2022 template<class T,class V> void _print(pair<T,V> p) {cerr<<"{";
    _print(p.first);cerr<<" "; _print(p.second);cerr<<"}";}
2023 template<class T>void _print(vector<T> v) {cerr<<" [ "; for (T i:v){_print(i);cerr<<"
    ";}cerr<<" ]";}
2024 template<class T>void _print(set<T> v) {cerr<<" [ "; for (T i:v){_print(i); cerr<<"
    ";}cerr<<" ]";}
2025 template<class T>void _print(multiset<T> v) {cerr<<" [ "; for (T
    i:v){_print(i);cerr<<" ";}cerr<<" ]";}
2026 template<class T,class V>void _print(map<T, V> v) {cerr<<" [ "; for(auto i:v)
    {_print(i);cerr<<" ";} cerr<<" ]";}
2027 typedef tree<ll, null_type, less<ll>, rb_tree_tag, tree_order_statistics_node_update>
    ordered_set;
2028 typedef tree<ll, null_type, less_equal<ll>, rb_tree_tag,
    tree_order_statistics_node_update> ordered_multiset;
2029 typedef tree<pair<ll,ll>, null_type, less<pair<ll,ll>>, rb_tree_tag,
    tree_order_statistics_node_update> ordered_pset;
2030 //-----
    -----
2031
2032
2033 const ll MOD=998244353;
2034
2035
2036 const ll MAX=5000300;
2037
2038 vector<ll> fact(MAX+2,1),inv_fact(MAX+2,1);
2039 ll binpow(ll a,ll b,ll MOD){
2040     ll ans=1;
2041     a%=MOD;
2042     while(b){
2043         if(b&1)
2044             ans=(ans*a)%MOD;
2045         b/=2;
2046         a=(a*a)%MOD;
2047     }
2048     return ans;
2049 }
2050 ll inverse(ll a,ll MOD){
2051     return binpow(a,MOD-2,MOD);
2052 }
2053 void precompute(ll MOD){
2054     for(ll i=2;i<MAX;i++){
2055         fact[i]=(fact[i-1]*i)%MOD;
2056     }
2057     inv_fact[MAX-1]=inverse(fact[MAX-1],MOD);
2058     for(ll i=MAX-2;i>=0;i--){
2059         inv_fact[i]=(inv_fact[i+1]*(i+1))%MOD;
2060     }
2061 }
2062 ll nCr(ll a,ll b,ll MOD){
2063     if((a<0)|| (a<b)|| (b<0))
2064         return 0;
2065     ll denom=(inv_fact[b]*inv_fact[a-b])%MOD;
2066     return (denom*fact[a])%MOD;
2067 }
2068 void solve(){
2069     ll n,a,b; cin>>n>>a>>b;
2070     ll ans=0;
2071     ll c=n-a-b;
2072
2073     assert(a>=2);
2074     for(ll i=b-c;i<a;i++){
2075         ll diff=b-c;
2076         if((diff&1) != (i&1)){
2077             continue;
2078         }
2079         ll l=(i+diff)/2,r=(i-diff)/2;
2080         ll now=nCr(i,l,MOD);

```

```

2081         ll lft=b+c-i;
2082         if(lft&1){
2083             continue;
2084         }
2085         lft/=2;
2086         ll ext=a-1-i;
2087         now=(now*nCr(lft-ext+a-2,a-2,MOD))%MOD;
2088         now=(now*binpow(2,ext,MOD))%MOD;
2089         now=(now*nCr(a-1,i,MOD))%MOD;
2090         ans+=now; ans%=MOD;
2091     }
2092     cout<<ans;
2093     return;
2094 }
2095 int main()
2096 {
2097     ios_base::sync_with_stdio(false);
2098     cin.tie(NULL);
2099     #ifndef ONLINE_JUDGE
2100     freopen("input.txt", "r", stdin);
2101     freopen("output.txt", "w", stdout);
2102     freopen("error.txt", "w", stderr);
2103     #endif
2104     ll test_cases=1;
2105     //cin>>test_cases;
2106     precompute(MOD);
2107     while(test_cases--){
2108         solve();
2109     }
2110     cout<<fixed<<setprecision(10);
2111     cerr<<"Time:"<<1000*((double)clock())/(double)CLOCKS_PER_SEC<<"ms\n";
2112 }
2113
2114 //JOINTXOR
2115 //Utkarsh.25dec
2116 #include <bits/stdc++.h>
2117 #define ll long long int
2118 #define pb push_back
2119 #define mp make_pair
2120 #define mod 1000000007
2121 #define vl vector<ll>
2122 #define all(c) (c).begin(),(c).end()
2123 using namespace std;
2124 ll power(ll a,ll b){ll res=1;a%=mod; assert(b>=0);
2125 for(;b;b>>=1){if(b&1)res=res*a%mod;a=a*a%mod;}return res;}
2126 ll modInverse(ll a){return power(a,mod-2);}
2127 const int N=500023;
2128 bool vis[N];
2129 vector<int> adj[N];
2130 long long readInt(long long l,long long r,char endd){
2131     long long x=0;
2132     int cnt=0;
2133     int fi=-1;
2134     bool is_neg=false;
2135     while(true){
2136         char g=getchar();
2137         if(g=='-'){
2138             assert(fi==-1);
2139             is_neg=true;
2140             continue;
2141         }
2142         if('0'<=g && g<='9'){
2143             x*=10;
2144             x+=g-'0';
2145             if(cnt==0){
2146                 fi=g-'0';
2147             }
2148             cnt++;
2149             assert(fi!=0 || cnt==1);

```



```

2149         assert(fi!=0 || is_neg==false);
2150
2151         assert(!(cnt>19 || ( cnt==19 && fi>1) ));
2152     } else if(g==endd){
2153         if(is_neg){
2154             x= -x;
2155         }
2156
2157         if(!(l <= x && x <= r))
2158         {
2159             cerr << l << ' ' << r << ' ' << x << '\n';
2160             assert(l == 0);
2161         }
2162
2163         return x;
2164     } else {
2165         assert(false);
2166     }
2167 }
2168 }
2169 string readString(int l,int r,char endd){
2170     string ret="";
2171     int cnt=0;
2172     while(true){
2173         char g=getchar();
2174         assert(g!=-1);
2175         if(g==endd){
2176             break;
2177         }
2178         cnt++;
2179         ret+=g;
2180     }
2181     assert(l<=cnt && cnt<=r);
2182     return ret;
2183 }
2184 long long readIntSp(long long l,long long r){
2185     return readInt(l,r,' ');
2186 }
2187 long long readIntLn(long long l,long long r){
2188     return readInt(l,r,'\n');
2189 }
2190 string readStringLn(int l,int r){
2191     return readString(l,r,'\n');
2192 }
2193 string readStringSp(int l,int r){
2194     return readString(l,r,' ');
2195 }
2196 int sumN=0;
2197 void solve()
2198 {
2199     // JOINTXOR
2200     int n=readInt(2,2000000,'\n');
2201     sumN+=n;
2202     assert(sumN<=2000000);
2203     string s=readString(n,n,'\n');
2204     int ones=0,zeros=0;
2205     for(int i=0;i<n;i++)
2206     {
2207         if(s[i]=='0')
2208             zeros++;
2209         else
2210             ones++;
2211         assert(s[i]=='0' || s[i]=='1');
2212     }
2213     if(zeros==0 || ones==0)
2214     {
2215         cout<<0<<'\n';
2216         return;
2217     }

```

```

2218     int st;
2219     for(int i=1;i<n;i++)
2220     {
2221         if(s[i]!=s[0])
2222         {
2223             st=i;
2224             break;
2225         }
2226     }
2227     int l2=st,r2=n-1;
2228     int diff=r2-l2;
2229     int nxt=0;
2230     for(int i=st+1;i<n;i++)
2231     {
2232         if(s[i]!=s[st])
2233         {
2234             nxt=i;
2235             break;
2236         }
2237     }
2238     int l1,r1;
2239     if(nxt==0)
2240         l1=0,r1=l1+diff;
2241     else
2242     {
2243         l1=l2-(nxt-l2);
2244         l1=max(l1,0);
2245         r1=l1+diff;
2246     }
2247     string maxi="";
2248     for(int i=0;i<=r1-l1;i++)
2249     {
2250         if(s[l1+i]!=s[l2+i])
2251             maxi+='1';
2252         else
2253             maxi+='0';
2254     }
2255     ll ans=0;
2256     for(int i=0;i<maxi.length();i++)
2257     {
2258         ans*=2;
2259         if(maxi[i]=='1')
2260             ans++;
2261         ans%=mod;
2262     }
2263     cout<<ans<<'\n';
2264 }
2265 int main()
2266 {
2267     #ifndef ONLINE_JUDGE
2268     freopen("input.txt", "r", stdin);
2269     freopen("output.txt", "w", stdout);
2270     #endif
2271     ios_base::sync_with_stdio(false);
2272     cin.tie(NULL),cout.tie(NULL);
2273     int T=readInt(1,100000,'\n');
2274     while(T--)
2275         solve();
2276     assert(getchar()==-1);
2277     cerr << "Time : " << 1000 * ((double)clock()) / (double)CLOCKS_PER_SEC << "ms\n";
2278 }
2279
2280 //PYTHAGORAS
2281 //Utkarsh.25dec
2282 #include <bits/stdc++.h>
2283 #define ll long long int
2284 #define pb push_back
2285 #define mp make_pair
2286 #define mod 1000000007

```

```

2287 #define vl vector <ll>
2288 #define all(c) (c).begin(), (c).end()
2289 using namespace std;
2290 ll power(ll a, ll b) {ll res=1; a%=mod; assert(b>=0);
2291   for(; b>=1; b>>=1) {if(b&1) res=res*a%mod; a=a*a%mod;} return res;}
2292 ll modInverse(ll a) {return power(a, mod-2);}
2293 const int N=500023;
2294 bool vis[N];
2295 vector <int> adj[N];
2296 long long readInt(long long l, long long r, char endd) {
2297   long long x=0;
2298   int cnt=0;
2299   int fi=-1;
2300   bool is_neg=false;
2301   while(true) {
2302     char g=getchar();
2303     if(g=='-') {
2304       assert(fi==-1);
2305       is_neg=true;
2306       continue;
2307     }
2308     if('0'<=g && g<='9') {
2309       x*=10;
2310       x+=g-'0';
2311       if(cnt==0) {
2312         fi=g-'0';
2313       }
2314       cnt++;
2315       assert(fi!=0 || cnt==1);
2316       assert(fi!=0 || is_neg==false);
2317
2318       assert(!(cnt>19 || ( cnt==19 && fi>1) ));
2319     } else if(g==endd) {
2320       if(is_neg) {
2321         x=-x;
2322       }
2323
2324       if(!(l <= x && x <= r))
2325       {
2326         cerr << l << ' ' << r << ' ' << x << '\n';
2327         assert(1 == 0);
2328       }
2329       return x;
2330     } else {
2331       assert(false);
2332     }
2333   }
2334 }
2335 string readString(int l, int r, char endd) {
2336   string ret="";
2337   int cnt=0;
2338   while(true) {
2339     char g=getchar();
2340     assert(g!=-1);
2341     if(g==endd) {
2342       break;
2343     }
2344     cnt++;
2345     ret+=g;
2346   }
2347   assert(l<=cnt && cnt<=r);
2348   return ret;
2349 }
2350 long long readIntSp(long long l, long long r) {
2351   return readInt(l, r, ' ');
2352 }
2353 long long readIntLn(long long l, long long r) {
2354   return readInt(l, r, '\n');

```

```

2355 }
2356 string readStringLn(int l,int r){
2357     return readString(l,r,'\n');
2358 }
2359 string readStringSp(int l,int r){
2360     return readString(l,r,' ');
2361 }
2362 int avail[N];
2363 pair<ll,ll> good[N];
2364 int issq[N];
2365 int sqrtval[N];
2366 void solve()
2367 {
2368     ll n=readInt(1,(ll)100000000*10000000,'\n');
2369     ll tmp=n;
2370     while(tmp%2==0)
2371         tmp/=2;
2372     assert(tmp<=100000);
2373     if(avail[tmp]==0)
2374         cout<<-1<<'\n';
2375     else
2376     {
2377         ll a=good[tmp].first;
2378         ll b=good[tmp].second;
2379         while(tmp!=n)
2380         {
2381             ll c=a+b;
2382             ll d=abs(a-b);
2383             a=c;
2384             b=d;
2385             tmp*=2;
2386         }
2387         cout<<a<<' '<<b<<'\n';
2388     }
2389 }
2390 int main()
2391 {
2392     #ifndef ONLINE_JUDGE
2393     freopen("input.txt", "r", stdin);
2394     freopen("output.txt", "w", stdout);
2395     #endif
2396     ios_base::sync_with_stdio(false);
2397     cin.tie(NULL),cout.tie(NULL);
2398     int T=readInt(1,100000,'\n');
2399     for(int i=0;i<400;i++)
2400     {
2401         issq[i*i]=1;
2402         sqrtval[i*i]=i;
2403     }
2404     for(int n=1;n<=100000;n++)
2405     {
2406         for(int a=0;a<=1000;a++)
2407         {
2408             if(a*a>n)
2409                 break;
2410             if(issq[n-a*a]==1)
2411             {
2412                 ll b=sqrtval[n-a*a];
2413                 good[n]=mp(a,b);
2414                 avail[n]=1;
2415                 break;
2416             }
2417         }
2418     }
2419     while(T--)
2420         solve();
2421     assert(getchar()==-1);
2422     cerr << "Time : " << 1000 * ((double)clock()) / (double)CLOCKS_PER_SEC << "ms\n";
2423 }

```

```

2424
2425 //STRANGE_BST
2426 #include <map>
2427 #include <set>
2428 #include <cmath>
2429 #include <ctime>
2430 #include <queue>
2431 #include <stack>
2432 #include <cstdio>
2433 #include <cstdlib>
2434 #include <vector>
2435 #include <cstring>
2436 #include <algorithm>
2437 using namespace std;
2438 typedef long long ll;
2439 const int N=100010;
2440 const int LOGN=28;
2441 int n;
2442 int lg2[N];
2443 ll S[N];
2444 int mn[N][LOGN],pos[N][LOGN];
2445
2446 struct nod
2447 {
2448     int w,g;
2449
2450     friend bool operator<(nod x,nod y)
2451     {
2452         return x.w<y.w;
2453     }
2454 }nd[N];
2455
2456 void init()
2457 {
2458     for(int i=1;i<=n;i++) S[i]=S[i-1]+nd[i].w;
2459     for(int i=1;i<=n;i++) lg2[i]=(int)log2(i);
2460     for(int i=1;i<=n;i++) mn[i][0]=nd[i].g-nd[i].w,pos[i][0]=i;
2461     for(int i=1;i<LOGN;i++)
2462     {
2463         for(int j=1;j<=n;j++)
2464         {
2465             int p=j+(1<<(i-1));
2466             if(p>n) mn[j][i]=mn[j][i-1],pos[j][i]=pos[j][i-1];
2467             else
2468             {
2469                 if(mn[j][i-1]<mn[p][i-1]) mn[j][i]=mn[j][i-1],pos[j][i]=pos[j][i-1];
2470                 else mn[j][i]=mn[p][i-1],pos[j][i]=pos[p][i-1];
2471             }
2472         }
2473     }
2474 }
2475
2476 int getmnpos(int L,int R)
2477 {
2478     int t=lg2[R-L+1];
2479     return mn[L][t]<mn[R-(1<<t)+1][t]?pos[L][t]:pos[R-(1<<t)+1][t];
2480 }
2481
2482 ll cal(int L,int R)
2483 {
2484     int x=getmnpos(L,R);
2485     ll ans=nd[x].g+S[x-1]-S[L-1]+S[R]-S[x];
2486     if(x!=L) ans=max(ans,cal(L,x-1));
2487     if(x!=R) ans=max(ans,cal(x+1,R));
2488     return ans;
2489 }
2490
2491 int main()
2492 {

```

```

2493     scanf("%d",&n);
2494     for(int i=1;i<=n;i++) scanf("%d",&nd[i].w);
2495     for(int i=1;i<=n;i++) scanf("%d",&nd[i].g);
2496     sort(nd+1,nd+n+1);
2497     init();
2498     printf("%lld\n",cal(1,n));
2499
2500     return 0;
2501 }
2502
2503 //GET_THE_SEG
2504 #include <map>
2505 #include <set>
2506 #include <cmath>
2507 #include <ctime>
2508 #include <queue>
2509 #include <stack>
2510 #include <cstdio>
2511 #include <cstdlib>
2512 #include <vector>
2513 #include <cstring>
2514 #include <algorithm>
2515 using namespace std;
2516 typedef double db;
2517 typedef long long ll;
2518 typedef unsigned long long ull;
2519 const int N=100000;
2520 const int LOGN=28;
2521 const ll TMD=0;
2522 const ll INF=2147483647;
2523 int T;
2524
2525 int query(int L,int R)
2526 {
2527     int tmp;
2528     printf("1 %d %d\n",L,R);
2529     fflush(stdout);
2530     scanf("%d",&tmp);
2531     return tmp;
2532 }
2533
2534 void answer(int L,int R)
2535 {
2536     int tmp;
2537     printf("2 %d %d\n",L,R);
2538     fflush(stdout);
2539     scanf("%d",&tmp);
2540 }
2541
2542 int main()
2543 {
2544     scanf("%d",&T);
2545     while(T--)
2546     {
2547         int n,k,L,R,M,pos1,pos2;
2548         scanf("%d%d",&n,&k);
2549         L=0;R=n+1;
2550         while(L+1!=R)
2551         {
2552             M=(L+R)>>1;
2553             if(query(1,M)==2*M) L=M;
2554             else R=M;
2555         }
2556         if(R==n+1) answer(1,k/2);
2557         else
2558         {
2559             pos1=R;
2560             if(query(pos1,n)>=k)
2561             {

```

```

2562         L=pos1;R=n+1;
2563         while(L+1!=R)
2564         {
2565             M=(L+R)>>1;
2566             if(query(pos1,M)<=k) L=M;
2567             else R=M;
2568         }
2569         if(query(pos1,L)==k) answer(pos1,L);
2570         else answer(pos1+1,L+1);
2571     }
2572     else
2573     {
2574         if((query(pos1,n)&1)^(k&1))
2575         {
2576             L=0;R=n+1;
2577             while(L+1!=R)
2578             {
2579                 M=(L+R)>>1;
2580                 if(query(M,n)==2*(n-M+1)) R=M;
2581                 else L=M;
2582             }
2583             pos2=L;
2584             answer(pos1-(k-query(pos1,pos2-1))/2,pos2-1);
2585         }
2586         else answer(pos1-(k-query(pos1,n))/2,n);
2587     }
2588 }
2589 }

```

```

2590
2591
2592     return 0;
2593 }
2594
2595 //DISJOINTXOR
2596 //Utkarsh.25dec
2597 #include <bits/stdc++.h>
2598 #define ll long long int
2599 #define pb push_back
2600 #define mp make_pair
2601 #define mod 1000000007
2602 #define vl vector<ll>
2603 #define all(c) (c).begin(),(c).end()
2604 using namespace std;
2605 ll power(ll a,ll b) {ll res=1;a%=mod; assert(b>=0);
2606 for(;b;b>>=1){if(b&1)res=res*a%mod;a=a*a%mod;}return res;}
2607 ll modInverse(ll a){return power(a,mod-2);}
2608 const int N=500023;
2609 bool vis[N];
2610 vector<int> adj[N];
2611 long long readInt(long long l,long long r,char endd){
2612     long long x=0;
2613     int cnt=0;
2614     int fi=-1;
2615     bool is_neg=false;
2616     while(true){
2617         char g=getchar();
2618         if(g=='-'){
2619             assert(fi==-1);
2620             is_neg=true;
2621             continue;
2622         }
2623         if('0'<=g && g<='9'){
2624             x*=10;
2625             x+=g-'0';
2626             if(cnt==0){
2627                 fi=g-'0';
2628             }
2629             cnt++;
2630             assert(fi!=0 || cnt==1);

```

```

2630         assert(fi!=0 || is_neg==false);
2631
2632         assert(!(cnt>19 || ( cnt==19 && fi>1) ));
2633     } else if(g==endd){
2634         if(is_neg){
2635             x= -x;
2636         }
2637
2638         if(!(l <= x && x <= r))
2639         {
2640             cerr << l << ' ' << r << ' ' << x << '\n';
2641             assert(l == 0);
2642         }
2643
2644         return x;
2645     } else {
2646         assert(false);
2647     }
2648 }
2649 }
2650 string readString(int l,int r,char endd){
2651     string ret="";
2652     int cnt=0;
2653     while(true){
2654         char g=getchar();
2655         assert(g!=-1);
2656         if(g==endd){
2657             break;
2658         }
2659         cnt++;
2660         ret+=g;
2661     }
2662     assert(l<=cnt && cnt<=r);
2663     return ret;
2664 }
2665 long long readIntSp(long long l,long long r){
2666     return readInt(l,r,' ');
2667 }
2668 long long readIntLn(long long l,long long r){
2669     return readInt(l,r,'\n');
2670 }
2671 string readStringLn(int l,int r){
2672     return readString(l,r,'\n');
2673 }
2674 string readStringSp(int l,int r){
2675     return readString(l,r,' ');
2676 }
2677 int sumN=0;
2678 void solve()
2679 {
2680     // DISJOINTXOR
2681     int n=readInt(2,5000,'\n');
2682     sumN+=(n*n);
2683     assert(sumN<=25000000);
2684     string s=readString(n,n,'\n');
2685     vector <int> ones,zeros;
2686     for(int i=0;i<n;i++)
2687     {
2688         assert(s[i]=='0' || s[i]=='1');
2689         if(s[i]=='1')
2690             ones.pb(i);
2691         else
2692             zeros.pb(i);
2693     }
2694     if(ones.size()==0 || zeros.size()==0)
2695     {
2696         cout<<0<<'\n';
2697         return;
2698     }

```



```

2699     int l=1,r=n/2;
2700     while(l<=r)
2701     {
2702         int mid=(l+r)/2;
2703         int flag=0;
2704         {
2705             int a=ones[0];
2706             auto it=lower_bound(all(zeros),a+mid);
2707             if(it!=zeros.end())
2708             {
2709                 if((*it)+mid-1<n)
2710                     flag=1;
2711             }
2712         }
2713         {
2714             int a=zeros[0];
2715             auto it=lower_bound(all(ones),a+mid);
2716             if(it!=ones.end())
2717             {
2718                 if((*it)+mid-1<n)
2719                     flag=1;
2720             }
2721         }
2722         if(flag)
2723             l=mid+1;
2724         else
2725             r=mid-1;
2726     }
2727     int len=r;
2728     vector <string> v;
2729     // Continuous segment of length 2*len
2730     for(int i=0;i<n;i++)
2731     {
2732         int l1=i,r1=i+len-1;
2733         int l2=r1+1,r2=l2+len-1;
2734         if(r2>=n)
2735             break;
2736         string maxi="";
2737         for(int i=0;i<=r1-l1;i++)
2738         {
2739             if(s[l1+i]!=s[l2+i])
2740                 maxi+='1';
2741             else
2742                 maxi+='0';
2743         }
2744         v.pb(maxi);
2745     }
2746     // Suffix of length len
2747     int r2=n-1,l2=r2-len+1;
2748     for(int i=0;i<n;i++)
2749     {
2750         int l1=i,r1=l1+len-1;
2751         if(r1>=l2)
2752             break;
2753         string maxi="";
2754         for(int i=0;i<=r1-l1;i++)
2755         {
2756             if(s[l1+i]!=s[l2+i])
2757                 maxi+='1';
2758             else
2759                 maxi+='0';
2760         }
2761         v.pb(maxi);
2762     }
2763     string ans=v[0];
2764     for(int i=1;i<v.size();i++)
2765         ans=max(ans,v[i]);
2766     ll out=0;
2767     for(int i=0;i<ans.length();i++)

```

```

2768     {
2769         out*=2;
2770         if(ans[i]=='1')
2771             out++;
2772         out%=mod;
2773     }
2774     cout<<out<<'\n';
2775 }
2776 int main()
2777 {
2778     #ifndef ONLINE_JUDGE
2779     freopen("input.txt", "r", stdin);
2780     freopen("output.txt", "w", stdout);
2781     #endif
2782     ios_base::sync_with_stdio(false);
2783     cin.tie(NULL),cout.tie(NULL);
2784     int T=readInt(1,100000,'\n');
2785     while(T--)
2786         solve();
2787     assert(getchar()==-1);
2788     cerr << "Time : " << 1000 * ((double)clock()) / (double)CLOCKS_PER_SEC << "ms\n";
2789 }
2790
2791 //WEIRDSUBARR
2792 #ifdef WTSH
2793     #include <wtsh.h>
2794 #else
2795     #include <bits/stdc++.h>
2796     using namespace std;
2797     #define dbg(...)
2798 #endif
2799
2800 #define int long long
2801 #define endl "\n"
2802 #define sz(w) (int)(w.size())
2803 using pii = pair<int, int>;
2804
2805 // ----- Input Checker Start -----
2806
2807 long long readInt(long long l, long long r, char endd)
2808 {
2809     long long x = 0;
2810     int cnt = 0, fi = -1;
2811     bool is_neg = false;
2812     while(true)
2813     {
2814         char g = getchar();
2815         if(g == '-')
2816         {
2817             assert(fi == -1);
2818             is_neg = true;
2819             continue;
2820         }
2821         if('0' <= g && g <= '9')
2822         {
2823             x *= 10;
2824             x += g - '0';
2825             if(cnt == 0)
2826                 fi = g - '0';
2827             cnt++;
2828             assert(fi != 0 || cnt == 1);
2829             assert(fi != 0 || is_neg == false);
2830             assert(!(cnt > 19 || (cnt == 19 && fi > 1)));
2831         }
2832         else if(g == endd)
2833         {
2834             if(is_neg)
2835                 x = -x;
2836             if(!(l <= x && x <= r))

```

```

2837         {
2838             cerr << "L: " << l << ", R: " << r << ", Value Found: " << x << '\n';
2839             assert(false);
2840         }
2841         return x;
2842     }
2843     else
2844     {
2845         assert(false);
2846     }
2847 }
2848 }
2849
2850 string readString(int l, int r, char endd)
2851 {
2852     string ret = "";
2853     int cnt = 0;
2854     while(true)
2855     {
2856         char g = getchar();
2857         assert(g != -1);
2858         if(g == endd)
2859             break;
2860         cnt++;
2861         ret += g;
2862     }
2863     assert(l <= cnt && cnt <= r);
2864     return ret;
2865 }
2866
2867 long long readIntSp(long long l, long long r) { return readInt(l, r, ' '); }
2868 long long readIntLn(long long l, long long r) { return readInt(l, r, '\n'); }
2869 string readStringSp(int l, int r) { return readString(l, r, ' '); }
2870 string readStringLn(int l, int r) { return readString(l, r, '\n'); }
2871 void readEOF() { assert(getchar() == EOF); }
2872
2873 vector<int> readVectorInt(int n, long long l, long long r)
2874 {
2875     vector<int> a(n);
2876     for(int i = 0; i < n - 1; i++)
2877         a[i] = readIntSp(l, r);
2878     a[n - 1] = readIntLn(l, r);
2879     return a;
2880 }
2881
2882 // ----- Input Checker End -----
2883
2884 int sumN = 0;
2885
2886 void solve()
2887 {
2888     int n = readIntLn(1, 1e5);
2889     sumN += n;
2890     vector<int> p = readVectorInt(n, 1, n);
2891     assert(set<int>(p.begin(), p.end()).size() == n);
2892     vector<int> hills{0};
2893     for(int i = 1; i + 1 < n; i++)
2894         if(p[i] > p[i - 1] and p[i] > p[i + 1])
2895             hills.push_back(i);
2896     hills.push_back(n - 1);
2897     int ans = 0;
2898     for(int i = 0; i + 1 < sz(hills); i++)
2899     {
2900         int len = hills[i + 1] - hills[i] + 1;
2901         ans += len * (len - 1) / 2;
2902     }
2903     ans += n;
2904     cout << ans << endl;
2905 }

```

```

2906
2907 int32_t main()
2908 {
2909     ios::sync_with_stdio(0);
2910     cin.tie(0);
2911     int T = readIntLn(1, 1e5);
2912     for(int tc = 1; tc <= T; tc++)
2913     {
2914         // cout << "Case #" << tc << ": ";
2915         solve();
2916     }
2917     assert(sumN <= 2e5);
2918     readEOF();
2919     return 0;
2920 }
2921
2922 //ROWBOMBING
2923 /**
2924  * the_hyp0cr1t3
2925  * 18.10.2022 18:59:48
2926  */
2927 #ifdef W
2928     #include <k_II.h>
2929 #else
2930     #include <bits/stdc++.h>
2931     using namespace std;
2932 #endif
2933
2934 // ----- Input Checker Start -----
2935
2936 long long readInt(long long l, long long r, char endd)
2937 {
2938     long long x = 0;
2939     int cnt = 0, fi = -1;
2940     bool is_neg = false;
2941     while(true)
2942     {
2943         char g = getchar();
2944         if(g == '-')
2945         {
2946             assert(fi == -1);
2947             is_neg = true;
2948             continue;
2949         }
2950         if('0' <= g && g <= '9')
2951         {
2952             x *= 10;
2953             x += g - '0';
2954             if(cnt == 0)
2955                 fi = g - '0';
2956             cnt++;
2957             assert(fi != 0 || cnt == 1);
2958             assert(fi != 0 || is_neg == false);
2959             assert(!(cnt > 19 || (cnt == 19 && fi > 1)));
2960         }
2961         else if(g == endd)
2962         {
2963             if(is_neg)
2964                 x = -x;
2965             if(!(l <= x && x <= r))
2966             {
2967                 cerr << "L: " << l << ", R: " << r << ", Value Found: " << x << '\n';
2968                 assert(false);
2969             }
2970             return x;
2971         }
2972         else
2973         {
2974             assert(false);

```

```

2975     }
2976 }
2977 }
2978
2979 string readString(int l, int r, char endd)
2980 {
2981     string ret = "";
2982     int cnt = 0;
2983     while(true)
2984     {
2985         char g = getchar();
2986         assert(g != -1);
2987         if(g == endd)
2988             break;
2989         cnt++;
2990         ret += g;
2991     }
2992     assert(l <= cnt && cnt <= r);
2993     return ret;
2994 }
2995
2996 long long readIntSp(long long l, long long r) { return readInt(l, r, ' '); }
2997 long long readIntLn(long long l, long long r) { return readInt(l, r, '\n'); }
2998 string readStringSp(int l, int r) { return readString(l, r, ' '); }
2999 string readStringLn(int l, int r) { return readString(l, r, '\n'); }
3000 void readEOF() { assert(getchar() == EOF); }
3001
3002 vector<int> readVectorInt(int n, long long l, long long r)
3003 {
3004     vector<int> a(n);
3005     for(int i = 0; i < n - 1; i++)
3006         a[i] = readIntSp(l, r);
3007     a[n - 1] = readIntLn(l, r);
3008     return a;
3009 }
3010
3011 // ----- Input Checker End -----
3012
3013 const pair<int, int> dirs[] = {{-1, 0}, {0, -1}, {1, 0}, {0, 1}};
3014 const int INF = 1e9;
3015
3016 auto chmax = [](auto& A, auto&& B) { return B > A? A = B, true : false; };
3017 auto chmin = [](auto& A, auto&& B) { return B < A? A = B, true : false; };
3018
3019 auto intended(int N, int M, vector<string> s) {
3020     vector d(N, vector<vector<int>>(M, vector<int>(2, INF)));
3021     deque<tuple<int, int, int, int>> q;
3022     q.emplace_back(0, 0, 0, 0); d[0][0][0] = 0;
3023     while(!q.empty()) {
3024         auto [x, y, o, dist] = q.front(); q.pop_front();
3025         if(d[x][y][o] < dist) continue;
3026
3027         if(chmin(d[x][y][1], dist + 1))
3028             q.emplace_back(x, y, 1, dist + 1);
3029
3030         for(auto [dx, dy]: dirs) {
3031             int nx = x + dx;
3032             int ny = y + dy;
3033             if(!(0 <= nx and nx < N and 0 <= ny and ny < M)) continue;
3034
3035             if(dx == 0) {
3036                 if(s[nx][ny] == '.' or o == 1) {
3037                     if(chmin(d[nx][ny][o], dist))
3038                         q.emplace_front(nx, ny, o, dist);
3039                 } else {
3040                     if(chmin(d[nx][ny][1], dist + 1))
3041                         q.emplace_back(nx, ny, 1, dist + 1);
3042                 }
3043             } else {

```

```

3044         if(s[nx][ny] == '.') {
3045             if(chmin(d[nx][ny][0], dist))
3046                 q.emplace_front(nx, ny, 0, dist);
3047         } else {
3048             if(chmin(d[nx][ny][1], dist + 1))
3049                 q.emplace_back(nx, ny, 1, dist + 1);
3050         }
3051     }
3052 }
3053 }
3054
3055 return min(d[N - 1][M - 1][0], d[N - 1][M - 1][1]);
3056 }
3057
3058 int main() {
3059     #if __cplusplus > 201703L
3060         namespace R = ranges;
3061     #endif
3062     ios_base::sync_with_stdio(false), cin.tie(nullptr);
3063     int64_t sum_n = 0;
3064
3065     int tests = readIntLn(1, 2e5);
3066     while(tests--) [&] {
3067         int N = readIntSp(1, 2e5);
3068         int M = readIntLn(1, 2e5);
3069         assert(1LL * N * M <= 5e5);
3070
3071         sum_n += 1LL * N * M;
3072         assert(sum_n <= 5e5);
3073
3074         vector<string> s(N);
3075         for(auto &x: s) {
3076             x = readStringLn(M, M);
3077             assert(count(x.begin(), x.end(), '.') + count(x.begin(), x.end(), '#') == M);
3078         }
3079
3080         cout << intended(N, M, s) << '\n';
3081     }();
3082
3083     cerr << sum_n << '\n';
3084
3085     #ifndef W
3086         readEOF();
3087     #endif
3088
3089 } // ~W
3090
3091 //ROWBOMBING-EDITOR
3092 #include "bits/stdc++.h"
3093 // #pragma GCC optimize("O3,unroll-loops")
3094 // #pragma GCC target("avx2,bmi,bmi2,lzcnt,popcnt")
3095 using namespace std;
3096 using ll = long long int;
3097 mt19937_64 rng(chrono::high_resolution_clock::now().time_since_epoch().count());
3098
3099 int main()
3100 {
3101     ios::sync_with_stdio(false); cin.tie(0);
3102
3103     int t; cin >> t;
3104     while (t--) {
3105         int n, m; cin >> n >> m;
3106         vector<string> grid(n);
3107         for (int i = 1; i <= n; ++i) {
3108             cin >> grid[i-1];
3109         }
3110         const int inf = 1e9;
3111         vector<vector<array<int, 2>>> dist(n, vector(m, array{inf, inf}));
3112         dist[0][0][0] = 0;

```

```

3113     deque<array<int, 3>> dq = {{0, 0, 0}};
3114     vector<array<int, 2>> dir = {{0,1}, {1, 0}, {-1, 0}, {0, -1}};
3115     auto upd = [&] (int x, int y, int lv, int d) {
3116         if (dist[x][y][lv] > d) {
3117             dist[x][y][lv] = d;
3118             dq.push_front({x, y, lv});
3119         }
3120     };
3121     while (!dq.empty()) {
3122         auto [x, y, lv] = dq.front(); dq.pop_front();
3123         int d = dist[x][y][lv];
3124         for (auto [dx, dy] : dir) {
3125             int nx = x + dx, ny = y + dy;
3126             if (nx < 0 or nx >= n or ny < 0 or ny >= m) continue;
3127             if (nx == x and (lv == 1 or grid[nx][ny] == '.')) upd(nx, ny, lv, d);
3128             else if (nx != x and grid[nx][ny] == '.') upd(nx, ny, 0, d);
3129         }
3130         for (int dx : {-1, 0, 1}) {
3131             if (x+dx < 0 or x+dx >= n) continue;
3132             if (dist[x+dx][y][1] > 1 + d) {
3133                 dist[x+dx][y][1] = 1 + d;
3134                 dq.push_back({x+dx, y, 1});
3135             }
3136         }
3137     }
3138     cout << min(dist[n-1][m-1][0], dist[n-1][m-1][1]) << '\n';
3139 }
3140 }
3141
3142 //SUBARRAYREM
3143 #ifdef WTSH
3144     #include <wtsh.h>
3145 #else
3146     #include <bits/stdc++.h>
3147     using namespace std;
3148     #define dbg(...)
3149 #endif
3150
3151 #define int long long
3152 #define endl "\n"
3153 #define sz(w) (int)(w.size())
3154 using pii = pair<int, int>;
3155
3156 // ----- Input Checker Start -----
3157
3158 long long readInt(long long l, long long r, char endd)
3159 {
3160     long long x = 0;
3161     int cnt = 0, fi = -1;
3162     bool is_neg = false;
3163     while(true)
3164     {
3165         char g = getchar();
3166         if(g == '-')
3167         {
3168             assert(fi == -1);
3169             is_neg = true;
3170             continue;
3171         }
3172         if('0' <= g && g <= '9')
3173         {
3174             x *= 10;
3175             x += g - '0';
3176             if(cnt == 0)
3177                 fi = g - '0';
3178             cnt++;
3179             assert(fi != 0 || cnt == 1);
3180             assert(fi != 0 || is_neg == false);
3181             assert(!(cnt > 19 || (cnt == 19 && fi > 1)));

```

```

3182     }
3183     else if(g == endd)
3184     {
3185         if(is_neg)
3186             x = -x;
3187         if(!(l <= x && x <= r))
3188         {
3189             cerr << "L: " << l << ", R: " << r << ", Value Found: " << x << '\n';
3190             assert(false);
3191         }
3192         return x;
3193     }
3194     else
3195     {
3196         assert(false);
3197     }
3198 }
3199 }
3200
3201 string readString(int l, int r, char endd)
3202 {
3203     string ret = "";
3204     int cnt = 0;
3205     while(true)
3206     {
3207         char g = getchar();
3208         assert(g != -1);
3209         if(g == endd)
3210             break;
3211         cnt++;
3212         ret += g;
3213     }
3214     assert(l <= cnt && cnt <= r);
3215     return ret;
3216 }
3217
3218 long long readIntSp(long long l, long long r) { return readInt(l, r, ' '); }
3219 long long readIntLn(long long l, long long r) { return readInt(l, r, '\n'); }
3220 string readStringSp(int l, int r) { return readString(l, r, ' '); }
3221 string readStringLn(int l, int r) { return readString(l, r, '\n'); }
3222 void readEOF() { assert(getchar() == EOF); }
3223
3224 vector<int> readVectorInt(int n, long long l, long long r)
3225 {
3226     vector<int> a(n);
3227     for(int i = 0; i < n - 1; i++)
3228         a[i] = readIntSp(l, r);
3229     a[n - 1] = readIntLn(l, r);
3230     return a;
3231 }
3232
3233 // ----- Input Checker End -----
3234
3235 int sumN = 0;
3236
3237 void solve()
3238 {
3239     int n = readIntLn(1, 1e5);
3240     sumN += n;
3241     vector<int> a = readVectorInt(n, 0, 1);
3242     int cnt0 = count(a.begin(), a.end(), 0);
3243     int cnt1 = count(a.begin(), a.end(), 1);
3244     int take = min(cnt0, cnt1);
3245     int ans = take;
3246     cnt0 -= take, cnt1 -= take;
3247     ans += cnt1 / 3;
3248     cout << ans << endl;
3249 }
3250

```



```

3251 int32_t main()
3252 {
3253     ios::sync_with_stdio(0);
3254     cin.tie(0);
3255     int T = readIntLn(1, 1e5);
3256     for(int tc = 1; tc <= T; tc++)
3257     {
3258         // cout << "Case #" << tc << ": ";
3259         solve();
3260     }
3261     assert(sumN <= 2e5);
3262     readEOF();
3263     return 0;
3264 }
3265
3266 //EQINV
3267 #ifndef WTSH
3268     #include <wtsh.h>
3269 #else
3270     #include <bits/stdc++.h>
3271     using namespace std;
3272     #define dbg(...)
3273 #endif
3274
3275 #define int long long
3276 #define endl "\n"
3277 #define sz(w) (int)(w.size())
3278 using pii = pair<int, int>;
3279
3280 const long long INF = 1e18;
3281
3282 const int N = 1e6 + 5;
3283
3284 // ----- Input Checker Start -----
3285
3286 long long readInt(long long l, long long r, char endd)
3287 {
3288     long long x = 0;
3289     int cnt = 0, fi = -1;
3290     bool is_neg = false;
3291     while(true)
3292     {
3293         char g = getchar();
3294         if(g == '-')
3295         {
3296             assert(fi == -1);
3297             is_neg = true;
3298             continue;
3299         }
3300         if('0' <= g && g <= '9')
3301         {
3302             x *= 10;
3303             x += g - '0';
3304             if(cnt == 0)
3305                 fi = g - '0';
3306             cnt++;
3307             assert(fi != 0 || cnt == 1);
3308             assert(fi != 0 || is_neg == false);
3309             assert(!(cnt > 19 || (cnt == 19 && fi > 1)));
3310         }
3311         else if(g == endd)
3312         {
3313             if(is_neg)
3314                 x = -x;
3315             if(!(l <= x && x <= r))
3316             {
3317                 cerr << "L: " << l << ", R: " << r << ", Value Found: " << x << '\n';
3318                 assert(false);
3319             }
3320         }
3321     }

```

```

3320         return x;
3321     }
3322     else
3323     {
3324         assert(false);
3325     }
3326 }
3327 }
3328
3329 string readString(int l, int r, char endd)
3330 {
3331     string ret = "";
3332     int cnt = 0;
3333     while(true)
3334     {
3335         char g = getchar();
3336         assert(g != -1);
3337         if(g == endd)
3338             break;
3339         cnt++;
3340         ret += g;
3341     }
3342     assert(l <= cnt && cnt <= r);
3343     return ret;
3344 }
3345
3346 long long readIntSp(long long l, long long r) { return readInt(l, r, ' '); }
3347 long long readIntLn(long long l, long long r) { return readInt(l, r, '\n'); }
3348 string readStringSp(int l, int r) { return readString(l, r, ' '); }
3349 string readStringLn(int l, int r) { return readString(l, r, '\n'); }
3350 void readEOF() { assert(getchar() == EOF); }
3351
3352 vector<int> readVectorInt(int n, long long l, long long r)
3353 {
3354     vector<int> a(n);
3355     for(int i = 0; i < n - 1; i++)
3356         a[i] = readIntSp(l, r);
3357     a[n - 1] = readIntLn(l, r);
3358     return a;
3359 }
3360
3361 // ----- Input Checker End -----
3362
3363 int sumN = 0, F = 0, S = 0;
3364
3365 void solve()
3366 {
3367     int n = readIntLn(1, 1e5);
3368     sumN += n;
3369     string a = readStringLn(n, n);
3370     string b = readStringLn(n, n);
3371     assert(*min_element(a.begin(), a.end()) >= '0' and *max_element(a.begin(), a.end())
3372            <= '1');
3373     assert(*min_element(b.begin(), b.end()) >= '0' and *max_element(b.begin(), b.end())
3374            <= '1');
3375     int dif = 0;
3376     for(int i = 0; i < n; i++)
3377         dif += a[i] != b[i];
3378     vector<pii> ops;
3379     if(dif <= (n + 1) / 2)
3380     {
3381         for(int i = 0; i < n; i++)
3382         {
3383             if(a[i] != b[i])
3384                 ops.push_back({1, i + 1});
3385         }
3386         F++;
3387     }
3388     else

```

```

3387     {
3388         for(int i = 0; i < n; i++)
3389         {
3390             if(a[i] == b[i])
3391                 ops.push_back({1, i + 1});
3392         }
3393         ops.push_back({2, -1});
3394         S++;
3395     }
3396     cout << sz(ops) << endl;
3397     for(auto &[type, i]: ops)
3398     {
3399         cout << type;
3400         if(type == 1)
3401             cout << " " << i;
3402         cout << endl;
3403     }
3404 }
3405
3406 int32_t main()
3407 {
3408     ios::sync_with_stdio(0);
3409     cin.tie(0);
3410     int T = readIntLn(1, 1e5);
3411     for(int tc = 1; tc <= T; tc++)
3412     {
3413         // cout << "Case #" << tc << ": ";
3414         solve();
3415     }
3416     readEOF();
3417     assert(sumN <= 2e5);
3418     cerr << "First type cases: " << F << endl;
3419     cerr << "Second type cases: " << S << endl;
3420     return 0;
3421 }
3422
3423 //ARRPAL
3424 #ifdef WTSH
3425     #include <wtsh.h>
3426 #else
3427     #include <bits/stdc++.h>
3428     using namespace std;
3429     #define dbg(...)
3430 #endif
3431
3432 #define int long long
3433 #define endl "\n"
3434 #define sz(w) (int)(w.size())
3435 using pii = pair<int, int>;
3436
3437 // ----- Input Checker Start -----
3438
3439 long long readInt(long long l, long long r, char endd)
3440 {
3441     long long x = 0;
3442     int cnt = 0, fi = -1;
3443     bool is_neg = false;
3444     while(true)
3445     {
3446         char g = getchar();
3447         if(g == '-')
3448         {
3449             assert(fi == -1);
3450             is_neg = true;
3451             continue;
3452         }
3453         if('0' <= g && g <= '9')
3454         {
3455             x *= 10;

```

```

3456         x += g - '0';
3457         if(cnt == 0)
3458             fi = g - '0';
3459         cnt++;
3460         assert(fi != 0 || cnt == 1);
3461         assert(fi != 0 || is_neg == false);
3462         assert(!(cnt > 19 || (cnt == 19 && fi > 1)));
3463     }
3464     else if(g == endd)
3465     {
3466         if(is_neg)
3467             x = -x;
3468         if(!(l <= x && x <= r))
3469         {
3470             cerr << "L: " << l << ", R: " << r << ", Value Found: " << x << '\n';
3471             assert(false);
3472         }
3473         return x;
3474     }
3475     else
3476     {
3477         assert(false);
3478     }
3479 }
3480 }
3481
3482 string readString(int l, int r, char endd)
3483 {
3484     string ret = "";
3485     int cnt = 0;
3486     while(true)
3487     {
3488         char g = getchar();
3489         assert(g != -1);
3490         if(g == endd)
3491             break;
3492         cnt++;
3493         ret += g;
3494     }
3495     assert(l <= cnt && cnt <= r);
3496     return ret;
3497 }
3498
3499 long long readIntSp(long long l, long long r) { return readInt(l, r, ' '); }
3500 long long readIntLn(long long l, long long r) { return readInt(l, r, '\n'); }
3501 string readStringSp(int l, int r) { return readString(l, r, ' '); }
3502 string readStringLn(int l, int r) { return readString(l, r, '\n'); }
3503 void readEOF() { assert(getchar() == EOF); }
3504
3505 vector<int> readVectorInt(int n, long long l, long long r)
3506 {
3507     vector<int> a(n);
3508     for(int i = 0; i < n - 1; i++)
3509         a[i] = readIntSp(l, r);
3510     a[n - 1] = readIntLn(l, r);
3511     return a;
3512 }
3513
3514 // ----- Input Checker End -----
3515
3516 int sumN = 0;
3517
3518 void solve()
3519 {
3520     int n = readIntLn(1, 3e5);
3521     sumN += n;
3522     vector<int> a = readVectorInt(n, 1, 1e9);
3523     vector<int> d;
3524     for(int i = 0; i <= n / 2 - 1; i++)

```

```

3525         d.push_back(a[n - 1 - i] - a[i]);
3526     if(*min_element(d.begin(), d.end()) >= 0 and is_sorted(d.rbegin(), d.rend()))
3527         cout << d[0] << endl;
3528     else
3529         cout << -1 << endl;
3530 }
3531
3532 int32_t main()
3533 {
3534     ios::sync_with_stdio(0);
3535     cin.tie(0);
3536     int T = readIntLn(1, 1e5);
3537     for(int tc = 1; tc <= T; tc++)
3538     {
3539         // cout << "Case #" << tc << ": ";
3540         solve();
3541     }
3542     assert(sumN <= 3e5);
3543     readEOF();
3544     return 0;
3545 }
3546
3547 //MAXIMUM_SUM
3548 #include <map>
3549 #include <set>
3550 #include <cmath>
3551 #include <ctime>
3552 #include <queue>
3553 #include <stack>
3554 #include <cstdio>
3555 #include <cstdlib>
3556 #include <vector>
3557 #include <cstring>
3558 #include <algorithm>
3559 using namespace std;
3560 typedef double db;
3561 typedef long long ll;
3562 typedef unsigned long long ull;
3563 const int N=1000010;
3564 const int LOGN=28;
3565 const ll TMD=10000000007;
3566 const ll INF=2147483647;
3567 int T,n;
3568 ll ans;
3569 int a[N];
3570 ll b[N];
3571 vector<int> v[N];
3572
3573 ll pw(ll x,ll p)
3574 {
3575     if(!p) return 1;
3576     ll y=pw(x,p>>1);
3577     y=(y*y)%TMD;
3578     if(p&1) y=(y*x)%TMD;
3579     return y;
3580 }
3581
3582
3583 int main()
3584 {
3585     scanf("%d",&T);
3586     while(T--)
3587     {
3588         scanf("%d",&n);
3589         for(int i=1;i<=n;i++) scanf("%d",&a[i]);
3590         for(int i=2;i<N;i++) v[i].clear();
3591         for(int i=1;i<=n;i++)
3592         {
3593             int t=a[i];

```

```

3594         for(int j=2;j*j<=t;j++)
3595         {
3596             if(t%j) continue;
3597             int cnt=0;
3598             while(!(t%j)) cnt++,t/=j;
3599             v[j].push_back(cnt);
3600         }
3601         if(t!=1) v[t].push_back(1);
3602     }
3603     for(int i=2;i<N;i++) sort(v[i].begin(),v[i].end(),greater<int>());
3604     for(int i=1;i<=n;i++) b[i]=1;
3605     for(int i=2;i<N;i++)
3606         for(int j=0;j<v[i].size();j++) b[j+1]=(b[j+1]*pw(i,v[i][j]))%TMD;
3607     ans=0;
3608     for(int i=1;i<=n;i++) ans=(ans+b[i])%TMD;
3609     printf("%lld\n",ans);
3610 }
3611
3612     return 0;
3613 }
3614
3615 //TRIXOR
3616 //Utkarsh.25dec
3617 #include <bits/stdc++.h>
3618 #define ll long long int
3619 #define pb push_back
3620 #define mp make_pair
3621 #define mod 1000000007
3622 #define vl vector<ll>
3623 #define all(c) (c).begin(),(c).end()
3624 using namespace std;
3625 ll power(ll a,ll b) {ll res=1;a%=mod; assert(b>=0);
3626 for(;b>=1){if(b&1)res=res*a%mod;a=a*a%mod;}return res;}
3627 ll modInverse(ll a){return power(a,mod-2);}
3628 const int N=500023;
3629 bool vis[N];
3630 vector<int> adj[N];
3631 long long readInt(long long l,long long r,char endd){
3632     long long x=0;
3633     int cnt=0;
3634     int fi=-1;
3635     bool is_neg=false;
3636     while(true){
3637         char g=getchar();
3638         if(g=='-'){
3639             assert(fi==-1);
3640             is_neg=true;
3641             continue;
3642         }
3643         if('0'<=g && g<='9'){
3644             x*=10;
3645             x+=g-'0';
3646             if(cnt==0){
3647                 fi=g-'0';
3648             }
3649             cnt++;
3650             assert(fi!=0 || cnt==1);
3651             assert(fi!=0 || is_neg==false);
3652
3653             assert(!(cnt>19 || ( cnt==19 && fi>1) ));
3654         } else if(g==endd){
3655             if(is_neg){
3656                 x= -x;
3657             }
3658
3659             if(!(l <= x && x <= r))
3660             {
3661                 cerr << l << ' ' << r << ' ' << x << '\n';
3662                 assert(1 == 0);

```

```

3662         }
3663
3664         return x;
3665     } else {
3666         assert(false);
3667     }
3668 }
3669
3670 string readString(int l,int r,char endd){
3671     string ret="";
3672     int cnt=0;
3673     while(true){
3674         char g=getchar();
3675         assert(g!=-1);
3676         if(g==endd){
3677             break;
3678         }
3679         cnt++;
3680         ret+=g;
3681     }
3682     assert(l<=cnt && cnt<=r);
3683     return ret;
3684 }
3685 long long readIntSp(long long l,long long r){
3686     return readInt(l,r,' ');
3687 }
3688 long long readIntLn(long long l,long long r){
3689     return readInt(l,r,'\n');
3690 }
3691 string readStringLn(int l,int r){
3692     return readString(l,r,'\n');
3693 }
3694 string readStringSp(int l,int r){
3695     return readString(l,r,' ');
3696 }
3697 void solve()
3698 {
3699     int n=readInt(6,1000,'\n');
3700     ll A[n+1]={0};
3701     for(int i=1;i<=n;i++){
3702     {
3703         if(i==n)
3704             A[i]=readInt(0,1000000000,'\n');
3705         else
3706             A[i]=readInt(0,1000000000,' ');
3707     }
3708     vector <tuple<ll,ll,ll>> opers;
3709     for(int bit=0;bit<=30;bit++){
3710     {
3711         set <int> indices;
3712         for(int i=1;i<=n;i++){
3713         {
3714             if((A[i]&(1<<bit))!=0)
3715                 indices.insert(i);
3716         }
3717         while(indices.size()>=3)
3718         {
3719             int p=(*indices.begin());
3720             indices.erase(p);
3721             int q=(*indices.begin());
3722             indices.erase(q);
3723             int r=(*indices.begin());
3724             indices.erase(r);
3725             ll P=A[p], Q=A[q], R=A[r];
3726             opers.pb(make_tuple(P, Q, R));
3727             A[p]=(P^Q);
3728             A[q]=(Q^R);
3729             A[r]=(R^P);
3730         }
3731     }

```

```

3731         if(indices.size()==1)
3732         {
3733             int p>(*indices.begin());
3734             vector<int> fun;
3735             fun.pb(p);
3736             for(int i=1;i<=n;i++)
3737             {
3738                 if(fun.size()==3)
3739                     break;
3740                 if(i==p)
3741                     continue;
3742                 fun.pb(i);
3743             }
3744             for(auto it:fun)
3745                 indices.insert(it);
3746             p=fun[0];
3747             int q=fun[1];
3748             int r=fun[2];
3749             ll P=A[p], Q=A[q], R=A[r];
3750            opers.pb(make_tuple(P, Q, R));
3751             A[p]=(P^Q);
3752             A[q]=(Q^R);
3753             A[r]=(R^P);
3754             indices.erase(q);
3755         }
3756         if(indices.size()==2)
3757         {
3758             int p>(*indices.begin());
3759             indices.erase(p);
3760             int q>(*indices.begin());
3761             vector<int> fun;
3762             fun.pb(p);
3763             for(int i=1;i<=n;i++)
3764             {
3765                 if(fun.size()==3)
3766                     break;
3767                 if(i==p)
3768                     continue;
3769                 if(i==q)
3770                     continue;
3771                 fun.pb(i);
3772             }
3773             p=fun[0];
3774             q=fun[1];
3775             int r=fun[2];
3776             ll P=A[p], Q=A[q], R=A[r];
3777            opers.pb(make_tuple(P, Q, R));
3778             A[p]=(P^Q);
3779             A[q]=(Q^R);
3780             A[r]=(R^P);
3781             indices.insert(p);
3782             indices.insert(r);
3783         }
3784         if(indices.size()==3)
3785         {
3786             int p>(*indices.begin());
3787             indices.erase(p);
3788             int q>(*indices.begin());
3789             indices.erase(q);
3790             int r>(*indices.begin());
3791             indices.erase(r);
3792             ll P=A[p], Q=A[q], R=A[r];
3793            opers.pb(make_tuple(P, Q, R));
3794             A[p]=(P^Q);
3795             A[q]=(Q^R);
3796             A[r]=(R^P);
3797         }
3798     }
3799     cout<<opers.size()<<"\n";

```



```

3800     for(auto it:opers)
3801         cout<<get<0>(it)<<' '<<get<1>(it)<<' '<<get<2>(it)<<'\n';
3802     }
3803 int main()
3804 {
3805     #ifndef ONLINE_JUDGE
3806     freopen("input.txt", "r", stdin);
3807     freopen("output.txt", "w", stdout);
3808     #endif
3809     ios_base::sync_with_stdio(false);
3810     cin.tie(NULL),cout.tie(NULL);
3811     int T=readInt(1,10,'\n');
3812     while(T--)
3813         solve();
3814     assert(getchar()==-1);
3815     cerr << "Time : " << 1000 * ((double)clock()) / (double)CLOCKS_PER_SEC << "ms\n";
3816 }
3817
3818 //REDGREEN
3819 #include <bits/stdc++.h>
3820 using namespace std;
3821 #define maxn 10007
3822 #define mod 998244353
3823
3824 long long int fact[maxn], ifact[maxn];
3825
3826 long long int mpow(long long int a, long long int b) {
3827     long long int res = 1;
3828     while(b) {
3829         if(b&1) res *= a, res %= mod;
3830         a *= a;
3831         a %= mod;
3832         b >>= 1;
3833     }
3834     return res;
3835 }
3836
3837 void pre() {
3838     fact[0] = fact[1] = ifact[0] = ifact[1] = 1;
3839     for(int i = 2; i < maxn; i++) fact[i] = fact[i - 1]*i, fact[i] %= mod;
3840     for(int i = 2; i < maxn; i++) ifact[i] = ifact[i - 1]*mpow(i, mod - 2), ifact[i] %=
3841     mod;
3842 }
3843
3844 long long int comb(long long int a, long long int b) {
3845     if(b == 0) return 1LL;
3846     long long int ans = fact[a];
3847     ans *= ifact[b];
3848     ans %= mod;
3849     ans *= ifact[a - b];
3850     ans %= mod;
3851     return ans;
3852 }
3853
3854 int main() {
3855     pre();
3856     int t;
3857     cin >> t;
3858     while(t--) {
3859         int n, m;
3860         cin >> n >> m;
3861         if((n + m - 1)&1) cout << "0\n";
3862         else {
3863             long long int paths = comb(n + m - 2, m - 1);
3864             paths *= ((fact[n + m - 1]*((ifact[(n + m - 1)/2]*ifact[(n + m -
3865             1)/2])%mod))%mod);
3866             paths %= mod;
3867             paths *= mpow(2, n*m - (n + m - 1));

```

```

3867         paths %= mod;
3868         cout << paths << "\n";
3869     }
3870 }
3871 return 0;
3872 }
3873
3874 //KBPER
3875 #include <bits/stdc++.h>
3876 using namespace std;
3877 #define  enl      '\n'
3878 #define  int      long long
3879 #define  sz(s)    (int)s.size()
3880 #define  all(v)   (v).begin(), (v).end()
3881
3882 mt19937 rng (chrono::high_resolution_clock::now().time_since_epoch().count());
3883 template <typename A, typename B> ostream& operator<< (ostream &cout, pair<A, B> const
&p) { return cout << "(" << p.first << ", " << p.second << ")"; }
3884 template <typename A, typename B> istream& operator>> (istream& cin, pair<A, B> &p)
{cin >> p.first; return cin >> p.second;}
3885 template <typename A> ostream& operator<< (ostream &cout, vector<A> const &v) {cout <<
 "["; for(int i = 0; i < v.size(); i++) {if (i) cout << ", "; cout << v[i];} return cout
<< "]" ;}
3886 template <typename A> istream& operator>> (istream& cin, vector<A> &x){for(int i = 0; i
< x.size()-1; i++) cin >> x[i]; return cin >> x[x.size()-1];}
3887 template <typename A, typename B> A amax (A &a, B b){ if (b > a) a = b ; return a; }
3888 template <typename A, typename B> A amin (A &a, B b){ if (b < a) a = b ; return a; }
3889
3890 const long long mod = 1e9+7;
3891 const long long inf = 1e18;
3892
3893 void solve() {
3894     int n,k;
3895     cin>>n>>k;
3896     vector<int>p(n);
3897     cin>>p;
3898
3899     int odd = 0,even = 0;
3900
3901     for(auto u:p) {
3902         if(u) {
3903             if(u&1) odd++;
3904             else even++;
3905         }
3906     }
3907
3908     auto find_odd = [&](int i) {
3909         int total = (n+1)/2;
3910         if(total < i+odd or i < 0) return 0LL;
3911         return total-i-odd;
3912     };
3913
3914     auto find_even = [&](int i) {
3915         int total = n/2;
3916         if(total < i+even) {
3917             return 0LL;
3918         }
3919         return total-i-even;
3920     };
3921
3922     vector<vector<vector<vector<int>>>>>dp(n+1,vector<vector<vector<int>>>>(n+1,vector<vect
or<int>>>(n+1,vector<int>(2)))));
3923     dp[0][0][0][0] = 1;
3924     dp[0][0][0][1] = 1;
3925     for(int i=1;i<=n;i++) {
3926         if(p[i-1]) {
3927             if(p[i-1]&1) odd--;
3928             else even--;

```

```

3929     }
3930     for(int j=0;j<=n;j++) {
3931         for(int k=1;k<=n;k++) {
3932             if(p[i-1] == 0) {
3933                 if(j) dp[i][j][k][0] =
3934                     find_even(j-1)*(dp[i-1][j-1][k][0]+dp[i-1][j-1][k-1][1])%mod;
3935                 dp[i][j][k][1] =
3936                     find_odd(i-1-j)*(dp[i-1][j][k-1][0]+dp[i-1][j][k][1])%mod;
3937             }
3938             else if(p[i-1]&1) {
3939                 dp[i][j][k][1] = (dp[i-1][j][k-1][0]+dp[i-1][j][k][1])%mod;
3940             }
3941             else {
3942                 if(j) dp[i][j][k][0] = (dp[i-1][j-1][k][0]+dp[i-1][j-1][k-1][1])%mod;
3943             }
3944         }
3945     }
3946     cout<<(dp[n][n/2][k][0]+dp[n][n/2][k][1])%mod<<endl;
3947 }
3948 signed main() {
3949     ios_base::sync_with_stdio(false);
3950     cin.tie(nullptr);cout.tie(nullptr);
3951     //freopen("./testCasesZip/input13.in","r",stdin);
3952     //freopen("./testCasesZip/input13.out","w",stdout);
3953     int testcases = 1;
3954     cin>>testcases;
3955     while(testcases--) solve();
3956     return 0;
3957 }
3958
3959 //SEVSEGFACT
3960 #include <bits/stdc++.h>
3961
3962 using namespace std;
3963
3964 const int MAX = 200007;
3965 const int MOD = 1000000007;
3966
3967 vector<long long> v;
3968
3969 bool div(string s, long long x) {
3970     long long curr = 0;
3971     for (char c : s) {
3972         curr *= 10; curr %= x;
3973         curr += (c - '0'); curr %= x;
3974     }
3975     return (curr == 0);
3976 }
3977
3978 void solve() {
3979     string n;
3980     cin >> n;
3981     int a, b, c, d, e, f, g;
3982     cin >> a >> b >> c >> d >> e >> f >> g;
3983     int dc[9];
3984     dc[1] = b + c, dc[2] = a + b + d + e + g, dc[5] = a + c + d + f + g, dc[6] = a + c
3985     + d + e + f + g;
3986     int res = dc[1];
3987     for (long long x : v) {
3988         if (div(n, x)) {
3989             int cost = 0;
3990             while (x) {
3991                 cost += dc[x % 10];
3992                 x /= 10;
3993             }
3994             res = min(res, cost);
3995         }
3996     }

```

```

3995     }
3996     cout << res << '\n';
3997 }
3998
3999 int main() {
4000     ios::sync_with_stdio(false);
4001     cin.tie(nullptr);
4002
4003     v.push_back(2);
4004     v.push_back(5);
4005     v.push_back(6);
4006     for (int i = 1; i <= 8; i++) {
4007         for (int x = 0; x < (1 << i); x++) {
4008             int k = x, cnt = 0;
4009             long long num = 0;
4010             for (int j = 0; j < i; j++) {
4011                 if (k & 1) {num += 5; cnt++;}
4012                 else {num += 6;}
4013                 k >>= 1;
4014                 num *= 10;
4015             }
4016             num += 6;
4017             if (cnt % 3 != 0) {v.push_back(num);}
4018         }
4019     }
4020     int tt; cin >> tt; for (int i = 1; i <= tt; i++) {solve();}
4021 }
4022
4023 //MAXSUBARR
4024 #include<bits/stdc++.h>
4025 using namespace std;
4026 #define lli long long
4027
4028 lli maxSubArraySum(vector<lli> a, int n)
4029 {
4030     lli max_tot = INT_MIN, m = 0;
4031
4032     for (int i = 0; i < n; i++) {
4033         m = m + a[i];
4034         if (max_tot < m)
4035             max_tot = m;
4036
4037         if (m < 0)
4038             m = 0;
4039     }
4040     return max_tot;
4041 }
4042
4043 int main(){
4044
4045     // freopen("output.txt","r",stdin);
4046     // freopen("output1.txt","w",stdout);
4047     int t;
4048     cin>>t;
4049     while(t--){
4050         int n,m;
4051         cin>>n;
4052         vector<lli> a(n);
4053         for(int i=0;i<n;i++){
4054             cin>>a[i];
4055         }
4056         cin>>m;
4057         vector<lli> b(m);
4058         lli p=0;
4059         for(int i=0;i<m;i++){
4060             cin>>b[i];
4061             if(b[i]>0)p+=b[i];
4062         }
4063         long long maxx;

```

```

4064
4065         a.insert(a.begin(),p);
4066         maxx=maxSubArraySum(a,n+1);
4067         a.erase(a.begin(),a.begin()+1);
4068         a.insert(a.begin()+n,p);
4069         maxx=max(maxx,maxSubArraySum(a,n+1));
4070         cout<<maxx<<endl;
4071     }
4072 }
4073
4074 //COLLINEAR
4075 #include <map>
4076 #include <set>
4077 #include <cmath>
4078 #include <ctime>
4079 #include <queue>
4080 #include <stack>
4081 #include <cstdio>
4082 #include <cstdlib>
4083 #include <vector>
4084 #include <cstring>
4085 #include <algorithm>
4086 using namespace std;
4087 typedef double db;
4088 typedef long long ll;
4089 typedef unsigned long long ull;
4090 const int N=1000010;
4091 const int LOGN=28;
4092 const ll TMD=0;
4093 const ll INF=2147483647;
4094 int n,q;
4095 int x[N],y[N];
4096
4097 struct nod1
4098 {
4099     int l,r,k,b;
4100     ll y;
4101     nod1 *lc,*rc;
4102 };
4103
4104 struct Segtree1
4105 {
4106     nod1 *root;
4107
4108     Segtree1()
4109     {
4110         build(&root,1,n);
4111     }
4112
4113     void newnod(nod1 **p,int L,int R)
4114     {
4115         *p=new(nod1);
4116         (*p)->l=L; (*p)->r=R; (*p)->k=(*p)->b=INF;
4117         if(L==R) (*p)->y=y[L];
4118     }
4119
4120     void build(nod1 **p,int L,int R)
4121     {
4122         newnod(p,L,R);
4123         if(L==R) return ;
4124         int M=(L+R)>>1;
4125         build(&(*p)->lc,L,M);
4126         build(&(*p)->rc,M+1,R);
4127     }
4128
4129     void pushdown(nod1 *p)
4130     {
4131         if(p->k==INF) return ;
4132         if(p->l!=p->r)

```

```

4133         {
4134             p->lc->k=p->rc->k=p->k;
4135             p->lc->b=p->rc->b=p->b;
4136         }
4137         if (p->l==p->r) p->y=(ll)p->k*x[p->l]+p->b;
4138         p->k=p->b=INF;
4139     }
4140
4141     void modify(int L,int R,int K,int B)
4142     {
4143         _modify(root,L,R,K,B);
4144     }
4145
4146     void _modify(nod1 *p,int L,int R,int K,int B)
4147     {
4148         pushdown(p);
4149         if (p->l==L&& p->r==R)
4150         {
4151             p->k=K;p->b=B;
4152             return ;
4153         }
4154         int M=(p->l+p->r)>>1;
4155         if (R<=M) _modify(p->lc,L,R,K,B);
4156         else if (L>M) _modify(p->rc,L,R,K,B);
4157         else
4158         {
4159             _modify(p->lc,L,M,K,B);
4160             _modify(p->rc,M+1,R,K,B);
4161         }
4162     }
4163
4164     ll query(int pos)
4165     {
4166         return _query(root,pos);
4167     }
4168
4169     ll _query(nod1 *p,int pos)
4170     {
4171         pushdown(p);
4172         if (p->l==p->r) return p->y;
4173         int M=(p->l+p->r)>>1;
4174         if (pos<=M) return _query(p->lc,pos);
4175         else return _query(p->rc,pos);
4176     }
4177 };
4178
4179 struct nod2
4180 {
4181     db    k,tag;
4182     int   l,r,equal;
4183     nod2 *lc,*rc;
4184 };
4185
4186 struct Segtree2
4187 {
4188     nod2 *root;
4189
4190     Segtree2()
4191     {
4192         build(&root,1,n-1);
4193     }
4194
4195     void newnod(nod2 **p,int L,int R)
4196     {
4197         *p=new(nod2);
4198         (*p)->l=L; (*p)->r=R; (*p)->tag=INF;
4199         if (L==R)
4200         {
4201             (*p)->equal=1;

```

```

4202         (*p)->k=(db) (y[L+1]-y[L]) / (db) (x[L+1]-x[L]);
4203     }
4204 }
4205
4206 void build(nod2 **p,int L,int R)
4207 {
4208     newnod(p,L,R);
4209     if(L==R) return ;
4210     int M=(L+R)>>1;
4211     build(&(*p)->lc,L,M);
4212     build(&(*p)->rc,M+1,R);
4213     (*p)->equal=((*p)->lc->equal&&(*p)->rc->equal&&(*p)->lc->k==(*p)->rc->k);
4214     (*p)->k=(*p)->lc->k;
4215 }
4216
4217 void pushdown(nod2 *p)
4218 {
4219     if(p->tag==INF) return ;
4220     if(p->l!=p->r) p->lc->tag=p->rc->tag=p->tag;
4221     p->equal=1;
4222     p->k=p->tag;
4223     p->tag=INF;
4224 }
4225
4226 void modify(int L,int R,db K)
4227 {
4228     _modify(root,L,R,K);
4229 }
4230
4231 void _modify(nod2 *p,int L,int R,db K)
4232 {
4233     pushdown(p);
4234     if(p->l==L&&p->r==R)
4235     {
4236         p->tag=K;
4237         return ;
4238     }
4239     int M=(p->l+p->r)>>1;
4240     if(R<=M) _modify(p->lc,L,R,K);
4241     else if(L>M) _modify(p->rc,L,R,K);
4242     else
4243     {
4244         _modify(p->lc,L,M,K);
4245         _modify(p->rc,M+1,R,K);
4246     }
4247     pushdown(p->lc);
4248     pushdown(p->rc);
4249     p->equal=(p->lc->equal&&p->rc->equal&&p->lc->k==p->rc->k);
4250     p->k=p->lc->k;
4251 }
4252
4253 int query(int L,int R)
4254 {
4255     return _query(root,L,R) != ((db) INF*INF);
4256 }
4257
4258 db _query(nod2 *p,int L,int R)
4259 {
4260     pushdown(p);
4261     if(p->l==L&&p->r==R) return p->equal?p->k:(db) INF*INF;
4262     int M=(p->l+p->r)>>1;
4263     if(R<=M) return _query(p->lc,L,R);
4264     else if(L>M) return _query(p->rc,L,R);
4265     else
4266     {
4267         db lk=_query(p->lc,L,M),rk=_query(p->rc,M+1,R);
4268         return lk==rk?lk:(db) INF*INF;
4269     }
4270 }

```

```

4271 };
4272
4273 int main()
4274 {
4275     scanf("%d%d", &n, &q);
4276     for(int i=1; i<=n; i++) scanf("%d%d", &x[i], &y[i]);
4277     Segtree1 TY;
4278     Segtree2 TK;
4279     for(int i=1; i<=q; i++)
4280     {
4281         int t, L, R, K, B;
4282         scanf("%d%d%d", &t, &L, &R);
4283         if(t==2)
4284         {
4285             if(R-L<=1) printf("YES\n");
4286             else printf("%s\n", TK.query(L, R-1)?"YES":"NO");
4287         }
4288         else
4289         {
4290             scanf("%d%d", &K, &B);
4291             TY.modify(L, R, K, B);
4292             if(L!=R) TK.modify(L, R-1, K);
4293             if(L!=1)
4294                 TK.modify(L-1, L-1, (db) (TY.query(L)-TY.query(L-1)) / (db) (x[L]-x[L-1]));
4295             if(R!=n) TK.modify(R, R, (db) (TY.query(R+1)-TY.query(R)) / (db) (x[R+1]-x[R]));
4296         }
4297     }
4298     return 0;
4299 }
4300 //COLLINEAR-EDITOR
4301 #include "bits/stdc++.h"
4302 // #pragma GCC optimize("O3,unroll-loops")
4303 // #pragma GCC target("avx2,bmi,bmi2,lzcnt,popcnt")
4304 using namespace std;
4305 using ll = long long int;
4306 mt19937_64 rng(chrono::high_resolution_clock::now().time_since_epoch().count());
4307
4308 const array<ll, 2> inf = {LLONG_MAX, LLONG_MAX};
4309 const array<ll, 2> minf = {LLONG_MIN, LLONG_MIN};
4310
4311 struct Data {
4312     array<ll, 2> mn = inf, mx = minf;
4313 };
4314 Data unit;
4315
4316 struct Node {
4317     using T = Data;
4318     T f(T a, T b) {
4319         a.mn = min(a.mn, b.mn);
4320         a.mx = max(a.mx, b.mx);
4321         return a;
4322     }
4323
4324     Node *l = 0, *r = 0;
4325     int lo, hi;
4326     T mset = unit;
4327     T val = unit;
4328     Node(int _lo, int _hi):lo(_lo),hi(_hi){}
4329     T query(int L, int R) {
4330         if (R <= lo || hi <= L) return unit;
4331         if (L <= lo && hi <= R) return val;
4332         push();
4333         return f(l->query(L, R), r->query(L, R));
4334     }
4335     void set(int L, int R, T x) {
4336         if (R <= lo || hi <= L) return;
4337         if (L <= lo && hi <= R) {
4338             mset = x;

```



```

4339         val = x;
4340     }
4341     else {
4342         push(), l->set(L, R, x), r->set(L, R, x);
4343         val = f(l->val, r->val);
4344     }
4345 }
4346 void push() {
4347     if (!l) {
4348         int mid = lo + (hi - lo)/2;
4349         l = new Node(lo, mid); r = new Node(mid, hi);
4350     }
4351     if (mset.mn != inf or mset.mx != minf)
4352         l->set(lo,hi,mset), r->set(lo,hi,mset), mset = unit;
4353 }
4354 };
4355
4356 const array<int, 2> unit2 = {INT_MIN, INT_MIN};
4357 struct Node2 {
4358     using T = array<int, 2>;
4359     T f(T a, T b) {
4360         if (a == unit2) return b;
4361         return a;
4362     }
4363
4364     Node2 *l = 0, *r = 0;
4365     int lo, hi;
4366     T mset = unit2;
4367     T val = unit2;
4368     Node2(int _lo,int _hi):lo(_lo),hi(_hi){}
4369     T query(int L, int R) {
4370         if (R <= lo || hi <= L) return unit2;
4371         if (L <= lo && hi <= R) return val;
4372         push();
4373         return f(l->query(L, R), r->query(L, R));
4374     }
4375     void set(int L, int R, T x) {
4376         if (R <= lo || hi <= L) return;
4377         if (L <= lo && hi <= R) {
4378             mset = x;
4379             val = x;
4380         }
4381         else {
4382             push(), l->set(L, R, x), r->set(L, R, x);
4383             val = f(l->val, r->val);
4384         }
4385     }
4386     void push() {
4387         if (!l) {
4388             int mid = lo + (hi - lo)/2;
4389             l = new Node2(lo, mid); r = new Node2(mid, hi);
4390         }
4391         if (mset != unit2)
4392             l->set(lo,hi,mset), r->set(lo,hi,mset), mset = unit2;
4393     }
4394 };
4395
4396 int main()
4397 {
4398     ios::sync_with_stdio(false); cin.tie(0);
4399
4400     int n, q; cin >> n >> q;
4401     vector<array<int, 2>> pt(n);
4402     for (auto &[x, y] : pt) cin >> x >> y;
4403     Node *slopes = new Node(0, n);
4404     Node2 *vals = new Node2(0, n);
4405     for (int i = 0; i+1 < n; ++i) {
4406         int num = pt[i+1][1] - pt[i][1];
4407         int den = pt[i+1][0] - pt[i][0];

```

```

4408         int g = gcd(num, den);
4409         num /= g, den /= g;
4410         if (den < 0) num *= -1, den *= -1;
4411         Data cur; cur.mn = cur.mx = array<ll, 2>{num, den};
4412         slopes -> set(i, i+1, cur);
4413     }
4414
4415     while (q-->0) {
4416         int type; cin >> type;
4417         if (type == 1) {
4418             int l, r, k, b; cin >> l >> r >> k >> b;
4419             Data cur; cur.mn = cur.mx = {k, 1};
4420             --l, --r;
4421             slopes -> set(l, r, cur);
4422             vals -> set(l, r+1, array{k, b});
4423
4424             auto upd = [&] (int ind) {
4425                 auto m1 = vals -> query(ind-1, ind);
4426                 auto m2 = vals -> query(ind, ind+1);
4427                 ll y1, y2;
4428                 if (m1 == unit2) y1 = pt[ind-1][1];
4429                 else y1 = 1LL*m1[0]*pt[ind-1][0] + m1[1];
4430                 if (m2 == unit2) y2 = pt[ind][1];
4431                 else y2 = 1LL*m2[0]*pt[ind][0] + m2[1];
4432
4433                 ll num = y2 - y1, den = pt[ind][0] - pt[ind-1][0];
4434                 ll g = gcd(num, den);
4435                 num /= g, den /= g;
4436                 if (den < 0) num *= -1, den *= -1;
4437                 Data cur; cur.mn = cur.mx = array<ll, 2>{num, den};
4438                 slopes -> set(ind-1, ind, cur);
4439             };
4440             if (l > 0) upd(l);
4441             if (r < n-1) upd(r+1);
4442         }
4443         else {
4444             int l, r; cin >> l >> r;
4445             if (l == r) {
4446                 cout << "Yes\n";
4447                 continue;
4448             }
4449             auto res = slopes -> query(l-1, r-1);
4450             if (res.mn == res.mx) cout << "Yes\n";
4451             else cout << "No\n";
4452         }
4453     }
4454 }
4455
4456 // Let s[i] = slope(i, i+1) = y[i+1] - y[i] / x[i+1] - x[i]
4457 // Query [L, R] = is s[L] = s[L+1] = ... = s[R-1]
4458 // upd(L, R, K, B) = ?
4459 // s[L] = s[L+1] = ... = s[R-1] = K
4460 // update s[L-1] and s[R] separately using y values
4461
4462 //SUSSTR
4463 #include <bits/stdc++.h>
4464 using namespace std;
4465
4466 int main() {
4467     int t; cin >> t;
4468     while (t-->0) {
4469         int n; cin >> n;
4470         string s; cin >> s;
4471         int L = 0, R = n-1;
4472         deque<char> dq;
4473         while (L <= R) {
4474             if (s[L] == '0') dq.push_front('0');
4475             else dq.push_back('1');
4476         }

```

```

4477         if (L < R) {
4478             if (s[R] == '1') dq.push_front('1');
4479             else dq.push_back('0');
4480         }
4481         ++L, --R;
4482     }
4483     for (auto c : dq) cout << c;
4484     cout << '\n';
4485 }
4486 }
4487
4488 //RANKLISTPAGE
4489 // Pratiyush Mishra
4490
4491 #include <bits/stdc++.h>
4492 #define ll long long int
4493 using namespace std;
4494
4495 void mainSolve()
4496 {
4497     int x;
4498     cin >> x;
4499     int ans = ((x + 24) / 25);
4500     cout << ans << endl;
4501 }
4502
4503 int main()
4504 {
4505     #ifndef ONLINE_JUDGE
4506         freopen("input.txt", "r", stdin);
4507         freopen("output.txt", "w", stdout);
4508     #endif
4509     int t;
4510     cin >> t;
4511     while (t--)
4512     {
4513         mainSolve();
4514     }
4515     return 0;
4516 }
4517
4518 //START58- EXTERNALSITE
4519 //START57
4520 //EVENSPLIT
4521 #include <bits/stdc++.h>
4522 #include <ext/pb_ds/assoc_container.hpp>
4523 #include <ext/pb_ds/tree_policy.hpp>
4524 // #include <sys/resource.h>
4525 #define double long double
4526 #define initrand mt19937 mt_rand(time(0));
4527 #define rand mt_rand()
4528 #define MOD 1000000007
4529 #define INF 1000000000
4530 #define mid(l, u) ((l+u)/2)
4531 #define rchild(i) (i*2 + 2)
4532 #define lchild(i) (i*2 + 1)
4533 #define mp(a, b) make_pair(a, b)
4534 #define lz lazup(l, u, i);
4535 #define ordered_set tree<pair<int, int>, null_type, less<pair<int, int>>,
rb_tree_tag, tree_order_statistics_node_update>
4536 using namespace std;
4537 using namespace __gnu_pbds;
4538 signed main(){
4539     ios_base::sync_with_stdio(false);
4540     cin.tie(NULL);
4541     cout.tie(NULL);
4542     int t;
4543     cin>>t;
4544     while(t--){

```

```

4545         int n;
4546         cin>>n;
4547         string s;
4548         cin>>s;
4549         if(n > 2) sort(s.begin(), s.end());
4550         cout<<s<<endl;
4551     }
4552 }
4553
4554 //TREEDIV
4555 #include <bits/stdc++.h>
4556 #include <ext/pb_ds/assoc_container.hpp>
4557 #include <ext/pb_ds/tree_policy.hpp>
4558 // #include <sys/resource.h>
4559 #define int long long
4560 #define double long double
4561 #define initrand mt19937 mt_rand(time(0));
4562 #define rand mt_rand()
4563 #define MOD 1000000007
4564 #define INF 1000000000
4565 #define mid(l, u) ((l+u)/2)
4566 #define rchild(i) (i*2 + 2)
4567 #define lchild(i) (i*2 + 1)
4568 #define mp(a, b) make_pair(a, b)
4569 #define lz lazup(l, u, i);
4570 #define ordered_set tree<pair<int, int>, null_type, less<pair<int, int>>,
rb_tree_tag, tree_order_statistics_node_update>
4571 using namespace std;
4572 using namespace __gnu_pbds;
4573 const int maxn = 3e4 + 5;
4574 int ans[maxn], A[maxn], sz[maxn], st[maxn], ft[maxn], ansd[(int)1e6 + 1], tt = 0,
currdiv = 1;;
4575 vector<int> adj[maxn], ver;
4576 vector<pair<int, int>> divisors[(int)1e6 + 1];
4577 bool notPrime[(int)1e6 + 1];
4578 map<int, int> currfac;
4579 int ex(int a, int b){
4580     if(b==0) return 1;
4581     int c = ex(a, b/2);
4582     c *= c;
4583     c %= MOD;
4584     if(b%2) c *= a;
4585     c %= MOD;
4586     return c;
4587 }
4588 int inverse(int x){
4589     return ex(x, MOD - 2);
4590 }
4591 int d(int a, int b){
4592     return (a*inverse(b))%MOD;
4593 }
4594 int m(int a, int b){
4595     return (a*b)%MOD;
4596 }
4597 void getsz(int v, int p){
4598     st[v] = tt++;
4599     ver.push_back(v);
4600     sz[v] = 1; // every vertex has itself in its subtree
4601     for(auto u : adj[v])
4602         if(u != p){
4603             getsz(u, v);
4604             sz[v] += sz[u]; // add size of child u to its parent(v)
4605         }
4606     ft[v] = tt;
4607 }
4608 void dfs(int v, int p, bool keep){
4609     int mx = -1, bigChild = -1;
4610     for(auto u : adj[v])
4611         if(u != p && sz[u] > mx)

```

```

4612         mx = sz[u], bigChild = u;
4613     for(auto u : adj[v])
4614         if(u != p && u != bigChild)
4615             dfs(u, v, 0); // run a dfs on small childs and clear them from cnt
4616     if(bigChild != -1)
4617         dfs(bigChild, v, 1); // bigChild marked as big and not cleared from cnt
4618     for(auto u : adj[v])
4619         if(u != p && u != bigChild)
4620             for(int p = st[u]; p < ft[u]; p++)
4621                 for(pair<int, int> j: divisors[A[ver[p]])]{
4622                     currdiv = d(currdiv, currfac[j.first] + 1);
4623                     currfac[j.first] += j.second;
4624                     currdiv = m(currdiv, currfac[j.first] + 1);
4625                 }
4626     for(pair<int, int> j: divisors[A[v]]){
4627         currdiv = d(currdiv, currfac[j.first] + 1);
4628         currfac[j.first] += j.second;
4629         currdiv = m(currdiv, currfac[j.first] + 1);
4630     }
4631     ans[v] = currdiv;
4632     //now cnt[c] is the number of vertices in subtree of vertex v that has color c. You
4633     //can answer the queries easily.
4634     if(keep == 0) {
4635         currdiv = 1;
4636         currfac.clear();
4637     }
4638     signed main(){
4639         ios_base::sync_with_stdio(false);
4640         cin.tie(NULL);
4641         cout.tie(NULL);
4642         for(int i = 1; i <= 1e6; i++) ansd[i] = 1;
4643         for(int i = 2; i <= 1e6; i++){
4644             if(notPrime[i]) continue;
4645             for(int j = i; j <= 1e6; j += i){
4646                 notPrime[j] = true;
4647                 int mult = 0;
4648                 int temp = j;
4649                 while((temp % i) == 0){
4650                     temp /= i;
4651                     mult++;
4652                 }
4653                 ansd[j] *= (mult + 1);
4654                 ansd[j] %= MOD;
4655                 divisors[j].push_back({i, mult});
4656             }
4657         }
4658         int t;
4659         cin >> t;
4660         while(t--){
4661             int n;
4662             cin >> n;
4663             ver.clear();
4664             currfac.clear();
4665             tt = 0;
4666             for(int i = 0; i < n; i++) adj[i].clear();
4667             for(int i = 0; i < n; i++) cin >> A[i];
4668             for(int i = 1; i < n; i++){
4669                 int u, v;
4670                 cin >> u >> v;
4671                 u--;
4672                 v--;
4673                 adj[u].push_back(v);
4674                 adj[v].push_back(u);
4675             }
4676             tt = 0;
4677             getsz(0, 0);
4678             currdiv = 1;
4679             dfs(0, 0, 9);

```

```

4680         for(int i = 0;i<n;i++){
4681             cout<<ans[i]<<" ";
4682         }
4683         cout<<endl;
4684     }
4685 }
4686
4687 //MAXEXP
4688 #include <bits/stdc++.h>
4689 #include <ext/pb_ds/assoc_container.hpp>
4690 #include <ext/pb_ds/tree_policy.hpp>
4691 // #include <sys/resource.h>
4692 #define double long double
4693 #define initrand mt19937 mt_rand(time(0));
4694 #define rand mt_rand()
4695 #define MOD 1000000007
4696 #define INF 10000000000
4697 #define mid(l, u) ((l+u)/2)
4698 #define rchild(i) (i*2 + 2)
4699 #define lchild(i) (i*2 + 1)
4700 #define mp(a, b) make_pair(a, b)
4701 #define lz lazup(l, u, i);
4702 #define ordered_set tree<pair<int, int>, null_type, less<pair<int, int>>,
rb_tree_tag, tree_order_statistics_node_update>
4703 using namespace std;
4704 using namespace __gnu_pbds;
4705 signed main(){
4706     ios_base::sync_with_stdio(false);
4707     cin.tie(NULL);
4708     cout.tie(NULL);
4709     int t;
4710     cin>>t;
4711     while(t--){
4712         int n;
4713         cin>>n;
4714         string s;
4715         cin>>s;
4716         vector<int> num;
4717         int pl = 0, mi = 0;
4718         for(int i = 0;i<n;i++){
4719             if(s[i] == '-') mi++;
4720             else if(s[i] == '+') pl++;
4721             else num.push_back(s[i] - '0');
4722         }
4723         sort(num.begin(), num.end());
4724         for(int i = num.size()-1;i>(pl + mi - 1);i--) cout<<num[i];
4725         for(int i = pl+mi-1;i>=0;i--){
4726             if(pl-->0) cout<<'+';
4727             else cout<<'-' ;
4728             cout<<num[i];
4729         }
4730         cout<<endl;
4731     }
4732 }
4733
4734 //DQUERY
4735 #include <bits/stdc++.h>
4736 #include <ext/pb_ds/assoc_container.hpp>
4737 #include <ext/pb_ds/tree_policy.hpp>
4738 // #include <sys/resource.h>
4739 #define double long double
4740 #define int long long
4741 #define initrand mt19937 mt_rand(time(0));
4742 #define rand mt_rand()
4743 #define MOD 1000000007
4744 #define INF 10000000000
4745 #define mid(l, u) ((l+u)/2)
4746 #define rchild(i) (i*2 + 2)
4747 #define lchild(i) (i*2 + 1)

```

```

4748 #define mp(a, b) make_pair(a, b)
4749 #define lz lazup(l, u, i);
4750 #define ordered_set tree<pair<int, int>, null_type, less<pair<int, int>>,
rb_tree_tag, tree_order_statistics_node_update>
4751 using namespace std;
4752 using namespace __gnu_pbds;
4753 bool notPrime[100001];
4754 vector<int> pfac[100001];
4755 signed main() {
4756     ios_base::sync_with_stdio(false);
4757     cin.tie(NULL);
4758     cout.tie(NULL);
4759     for(int i = 2; i <= 100000; i++) {
4760         if(notPrime[i]) continue;
4761         for(int j = i; j <= 1e5; j += i) {
4762             notPrime[j] = true;
4763             pfac[j].push_back(i);
4764         }
4765     }
4766     int t;
4767     cin >> t;
4768     while(t--) {
4769         int n;
4770         cin >> n;
4771         int m[n];
4772         for(int i = 0; i < n; i++) cin >> m[i];
4773         int pref[n];
4774         pref[0] = m[0];
4775         for(int i = 1; i < n; i++) pref[i] = m[i] + pref[i-1];
4776         vector<int> li[100001];
4777         for(int i = 0; i < n; i++) {
4778             for(int j: pfac[m[i]]) {
4779                 li[j].push_back(i);
4780             }
4781         }
4782         vector<int> pf[100001], sf[100001];
4783         for(int i = 0; i <= 100000; i++) {
4784             if(li[i].size() == 0) continue;
4785             pf[i].push_back(m[li[i][0]]);
4786             for(int j = 1; j < li[i].size(); j++) {
4787                 pf[i].push_back(m[li[i][j]] + pf[i][j-1]);
4788             }
4789             vector<int> temp;
4790             for(int j: li[i]) temp.push_back(m[j]);
4791             sort(temp.begin(), temp.end());
4792             sf[i].push_back(temp[temp.size() - 1]);
4793             for(int j = 1; j < temp.size(); j++) {
4794                 sf[i].push_back(temp[temp.size() - 1 - j] + sf[i][j-1]);
4795             }
4796         }
4797         int q;
4798         cin >> q;
4799         while(q--) {
4800             int p, k;
4801             cin >> p >> k;
4802             int ans = pref[k-1];
4803             int cnt = lower_bound(li[p].begin(), li[p].end(), k) - li[p].begin();
4804             if(cnt > 0) {
4805                 ans -= pf[p][cnt - 1];
4806                 ans += sf[p][cnt - 1];
4807             }
4808             cout << ans << endl;
4809         }
4810     }
4811 }
4812 }
4813
4814 //SUMNEQ
4815 #include <bits/stdc++.h>

```

```

4816 #include <ext/pb_ds/assoc_container.hpp>
4817 #include <ext/pb_ds/tree_policy.hpp>
4818 // #include <sys/resource.h>
4819 #define double long double
4820 #define initrand mt19937 mt_rand(time(0));
4821 #define rand mt_rand()
4822 #define MOD 1000000007
4823 #define INF 10000000000
4824 #define mid(l, u) ((l+u)/2)
4825 #define rchild(i) (i*2 + 2)
4826 #define lchild(i) (i*2 + 1)
4827 #define mp(a, b) make_pair(a, b)
4828 #define lz lazup(l, u, i);
4829 #define ordered_set tree<pair<int, int>, null_type, less<pair<int, int>>,
rb_tree_tag, tree_order_statistics_node_update>
4830 using namespace std;
4831 using namespace __gnu_pbds;
4832 signed main(){
4833     ios_base::sync_with_stdio(false);
4834     cin.tie(NULL);
4835     cout.tie(NULL);
4836     int n;
4837     cin>>n;
4838     cout<<(n-1)<<endl;
4839 }
4840
4841 //NONNEGPROD
4842 #include <bits/stdc++.h>
4843 #include <ext/pb_ds/assoc_container.hpp>
4844 #include <ext/pb_ds/tree_policy.hpp>
4845 // #include <sys/resource.h>
4846 #define double long double
4847 #define int long long
4848 #define initrand mt19937 mt_rand(time(0));
4849 #define rand mt_rand()
4850 #define MOD 1000000007
4851 #define INF 10000000000
4852 #define mid(l, u) ((l+u)/2)
4853 #define rchild(i) (i*2 + 2)
4854 #define lchild(i) (i*2 + 1)
4855 #define mp(a, b) make_pair(a, b)
4856 #define lz lazup(l, u, i);
4857 #define ordered_set tree<pair<int, int>, null_type, less<pair<int, int>>,
rb_tree_tag, tree_order_statistics_node_update>
4858 using namespace std;
4859 using namespace __gnu_pbds;
4860 signed main(){
4861     ios_base::sync_with_stdio(false);
4862     cin.tie(NULL);
4863     cout.tie(NULL);
4864     int t;
4865     cin>>t;
4866     while(t--) {
4867         int n;
4868         cin>>n;
4869         int a[n];
4870         int ans = 0;
4871         bool z = 0;
4872         for(int i = 0; i<n; i++){
4873             cin>>a[i];
4874             ans = ans ^ (a[i] < 0);
4875             z = z || (a[i] == 0);
4876         }
4877         cout<<(ans && !z)<<endl;
4878     }
4879 }
4880
4881
4882 //TWODIFFPALIN

```



```

4883 #include <iostream>
4884 using namespace std;
4885
4886 int main() {
4887     int t;
4888     cin>>t;
4889     while(t--){
4890         int a, b;
4891         cin>>a>>b;
4892         if(a%2 && b%2){
4893             cout<<"No\n";
4894         }
4895         else if(min(a, b) == 1){
4896             cout<<"No\n";
4897         }
4898         else{
4899             cout<<"Yes\n";
4900         }
4901     }
4902     return 0;
4903 }
4904
4905 //COLOUR
4906
4907 //Utkarsh.25dec
4908 #include <bits/stdc++.h>
4909 #define ll long long int
4910 #define pb push_back
4911 #define mp make_pair
4912 #define mod 1000000007
4913 #define vl vector <ll>
4914 #define all(c) (c).begin(), (c).end()
4915 using namespace std;
4916 ll power(ll a,ll b) {ll res=1;a%=mod; assert(b>=0);
4917 for(;b>=1){if(b&1) res=res*a%mod;a=a*a%mod;}return res;}
4918 ll modInverse(ll a){return power(a,mod-2);}
4919 const int N=500023;
4920 bool vis[N];
4921 vector <int> adj[N];
4922 long long readInt(long long l,long long r,char endd){
4923     long long x=0;
4924     int cnt=0;
4925     int fi=-1;
4926     bool is_neg=false;
4927     while(true){
4928         char g=getchar();
4929         if(g=='-'){
4930             assert(fi==-1);
4931             is_neg=true;
4932             continue;
4933         }
4934         if('0'<=g && g<='9'){
4935             x*=10;
4936             x+=g-'0';
4937             if(cnt==0){
4938                 fi=g-'0';
4939             }
4940             cnt++;
4941             assert(fi!=0 || cnt==1);
4942             assert(fi!=0 || is_neg==false);
4943
4944             assert(!(cnt>19 || ( cnt==19 && fi>1) ));
4945         } else if(g==endd){
4946             if(is_neg){
4947                 x= -x;
4948             }
4949
4950             if(!(l <= x && x <= r))
4951                 {

```

```

4951         cerr << l << ' ' << r << ' ' << x << '\n';
4952         assert(1 == 0);
4953     }
4954
4955     return x;
4956 } else {
4957     assert(false);
4958 }
4959 }
4960 }
4961 string readString(int l,int r,char endd){
4962     string ret="";
4963     int cnt=0;
4964     while(true){
4965         char g=getchar();
4966         assert(g!=-1);
4967         if(g==endd){
4968             break;
4969         }
4970         cnt++;
4971         ret+=g;
4972     }
4973     assert(l<=cnt && cnt<=r);
4974     return ret;
4975 }
4976 long long readIntSp(long long l,long long r){
4977     return readInt(l,r,' ');
4978 }
4979 long long readIntLn(long long l,long long r){
4980     return readInt(l,r,'\n');
4981 }
4982 string readStringLn(int l,int r){
4983     return readString(l,r,'\n');
4984 }
4985 string readStringSp(int l,int r){
4986     return readString(l,r,' ');
4987 }
4988 void solve()
4989 {
4990     vector<int> a(3);
4991     a[0]=readInt(0,100,' ');
4992     a[1]=readInt(0,100,' ');
4993     a[2]=readInt(0,100,'\n');
4994
4995     sort(a.rbegin(), a.rend());
4996     int ans = 0;
4997
4998     for(int i = 0; i<3; i++) {
4999         if(a[i]){
5000             ans++;
5001             a[i]--;
5002         }
5003     }
5004
5005     for(int i = 0; i<3; i++) {
5006         for(int j = i+1; j<3; j++) {
5007             if(a[i] && a[j]) {
5008                 ans++;
5009                 a[i]--;
5010                 a[j]--;
5011             }
5012         }
5013     }
5014     cout<<ans<<endl;
5015 }
5016 int main()
5017 {
5018     #ifndef ONLINE_JUDGE
5019     freopen("input.txt", "r", stdin);

```

```

5020     freopen("output.txt", "w", stdout);
5021     #endif
5022     ios_base::sync_with_stdio(false);
5023     cin.tie(NULL), cout.tie(NULL);
5024     int T=readInt(1,100000,'\n');
5025     while(T--)
5026         solve();
5027     assert(getchar() == -1);
5028     cerr << "Time : " << 1000 * ((double)clock()) / (double)CLOCKS_PER_SEC << "ms\n";
5029 }
5030
5031 //KSUB
5032 #include <bits/stdc++.h>
5033 using namespace std;
5034
5035 int gcd (int a, int b) {
5036     if (b == 0)
5037         return a;
5038     else
5039         return gcd (b, a % b);
5040 }
5041
5042 int main(){
5043
5044     int t;
5045     cin >> t;
5046
5047     while(t--){
5048
5049         int n, k;
5050         cin >> n >> k;
5051
5052         vector <int> v(n);
5053         int G = 0;
5054
5055         for(int i = 0; i < n; i++){
5056             cin >> v[i];
5057             G = gcd(G, v[i]);
5058         }
5059
5060         int currG = 0, count = 0;
5061
5062         for(int i = 0; i < n; i++){
5063             currG = gcd(currG, v[i]);
5064             if(currG == G){
5065                 count++;
5066                 currG = 0;
5067             }
5068             if(count == k) break;
5069         }
5070
5071         if(count == k){
5072             cout << "YES\n";
5073         }
5074         else{
5075             cout << "NO\n";
5076         }
5077     }
5078     return 0;
5079 }
5080
5081 //SUBSBIN
5082 #include <bits/stdc++.h>
5083 using namespace std;
5084
5085 int main() {
5086     //freopen("inp3.in", "r", stdin);
5087     //freopen("out3.txt", "w", stdout);

```

```

5089     int t;
5090     cin >> t;
5091     while(t--) {
5092         int n;
5093         cin >> n;
5094         string s;
5095         cin >> s;
5096         int c[2] = {0, 0};
5097         for(int i = 0; i < s.size(); i++) c[s[i] - '0']++;
5098         if(c[0] && c[1]) {
5099             cout << "1 " << max(c[0], c[1]) - min(c[0], c[1]) + 1 << "\n";
5100             while(c[0] != c[1]) {
5101                 string now = "";
5102                 int flag = 1;
5103                 for(int i = 0; i < s.size(); i++) {
5104                     if(s[i] != s[i + 1] && flag) {
5105                         flag = 0;
5106                         cout << i + 1 << " " << i + 2 << " ";
5107                         if(c[0] > c[1]) cout << "1\n", c[1]++, now += '1';
5108                         else cout << "0\n", c[0]++, now += '0';
5109                         i++;
5110                     } else now += s[i];
5111                 }
5112                 s = now;
5113             }
5114             cout << "1 " << s.size() << " 0\n";
5115         } else cout << s.size() << " 0\n";
5116     }
5117 }
5118
5119 //ENODE_HARD
5120 #include <map>
5121 #include <set>
5122 #include <cmath>
5123 #include <ctime>
5124 #include <queue>
5125 #include <stack>
5126 #include <cstdio>
5127 #include <cstdlib>
5128 #include <vector>
5129 #include <cstring>
5130 #include <algorithm>
5131 using namespace std;
5132 typedef double db;
5133 typedef long long ll;
5134 typedef unsigned long long ull;
5135 const int N=1000010;
5136 const int LOGN=28;
5137 const ll TMD=0;
5138 const ll INF=2147483647;
5139 int n,q,cur;
5140 int a[N],key[N],tag[N],enough[N],tot[N];
5141 ll ans[N];
5142 vector<int> val;
5143 vector<int> G[N];
5144
5145 struct query
5146 {
5147     int t,id;
5148     ll k;
5149
5150     query() {}
5151
5152     query(int t,ll k,int id):t(t),k(k),id(id) {}
5153 };
5154 vector<query> Q[N];
5155
5156 void DFS(int x,int pre)
5157 {

```

```

5158     tot[x]=tot[pre]+enough[x];
5159     for(int i=0;i<G[x].size();i++)
5160     {
5161         int y=G[x][i];
5162         if(y==pre) continue;
5163         DFS(y,x);
5164     }
5165 }
5166
5167 void init()
5168 {
5169     scanf("%d",&n);
5170     for(int i=1;i<=n;i++) scanf("%d",&a[i]);
5171     for(int i=1;i<n;i++)
5172     {
5173         int u,v;
5174         scanf("%d%d",&u,&v);
5175         G[u].push_back(v);
5176         G[v].push_back(u);
5177     }
5178     scanf("%d",&q);
5179     for(int i=1;i<=q;i++)
5180     {
5181         int x,t;
5182         ll k;
5183         scanf("%d%d%lld",&x,&t,&k);
5184         Q[x].push_back(query(t,k,i));
5185     }
5186 }
5187
5188 void solve()
5189 {
5190     for(int i=1;i<=q;i++) ans[i]=-1;
5191     for(int i=1;i<=n;i++)
5192     {
5193         key[i]+=a[i];
5194         for(int j=0;j<G[i].size();j++) key[i]+=a[G[i][j]];
5195         if(!tag[key[i]]&&key[i]) tag[key[i]]=1,val.push_back(key[i]);
5196     }
5197     sort(val.begin(),val.end(),greater<int>());
5198     for(int i=0;i<val.size();i++)
5199     {
5200         for(int j=1;j<=n;j++) if(key[j]>=val[i]) enough[j]=1;
5201         DFS(1,0);
5202         for(int j=1;j<=n;j++)
5203         {
5204             for(int k=0;k<Q[j].size();k++)
5205             {
5206                 query qu=Q[j][k];
5207                 if(ans[qu.id]==-1&&tot[j]>=qu.t)
5208                 {
5209                     ll L=0,R=(ll)sqrt(qu.k/val[i]*2+1)+2,M;
5210                     while(L+1!=R)
5211                     {
5212                         M=(L+R)>>1;
5213                         if(M*(1+M)/2*val[i]>=qu.k) R=M;
5214                         else L=M;
5215                     }
5216                     ans[qu.id]=R;
5217                 }
5218             }
5219         }
5220     }
5221     for(int i=1;i<=q;i++) printf("%lld\n",ans[i]);
5222 }
5223
5224 int main()
5225 {
5226     init();

```

```

5227     solve();
5228
5229     return 0;
5230 }
5231
5232 //ENODE_HARD - EDITOR
5233 #include "bits/stdc++.h"
5234 // #pragma GCC optimize("O3,unroll-loops")
5235 // #pragma GCC target("avx2,bmi,bmi2,lzcnt,popcnt")
5236 using namespace std;
5237 using ll = long long int;
5238 mt19937_64 rng(chrono::high_resolution_clock::now().time_since_epoch().count());
5239
5240 struct query {
5241     ll t, k, id;
5242 };
5243
5244 /**
5245  * Point-update Segment Tree
5246  * Source: kactl
5247  * Description: Iterative point-update segment tree, ranges are half-open i.e [L, R).
5248  *              f is any associative function.
5249  * Time: O(logn) update/query
5250  */
5251
5252 template<class T, T unit = T()>
5253 struct SegTree {
5254     T f(T a, T b) { return a+b; }
5255     vector<T> s; int n;
5256     SegTree(int _n = 0, T def = unit) : s(2*_n, def), n(_n) {}
5257     void update(int pos, T val) {
5258         for (s[pos += n] += val; pos /= 2;)
5259             s[pos] = f(s[pos * 2], s[pos * 2 + 1]);
5260     }
5261     T query(int b, int e) {
5262         T ra = unit, rb = unit;
5263         for (b += n, e += n; b < e; b /= 2, e /= 2) {
5264             if (b % 2) ra = f(ra, s[b++]);
5265             if (e % 2) rb = f(s[--e], rb);
5266         }
5267         return f(ra, rb);
5268     }
5269 };
5270
5271 int main()
5272 {
5273     ios::sync_with_stdio(false); cin.tie(0);
5274
5275     int n; cin >> n;
5276     vector<int> a(n), b(n);
5277     for (int &x : a) cin >> x;
5278     vector<vector<int>> g(n);
5279     for (int i = 0; i < n-1; ++i) {
5280         int u, v; cin >> u >> v;
5281         g[--u].push_back(--v);
5282         g[v].push_back(u);
5283     }
5284     for (int i = 0; i < n; ++i) {
5285         b[i] = a[i];
5286         for (int u : g[i]) b[i] += a[u];
5287     }
5288     vector<vector<query>> queries(n);
5289     int q; cin >> q;
5290     for (int i = 0; i < q; ++i) {
5291         query cur;
5292         int u; cin >> u; --u;
5293         cin >> cur.t >> cur.k;
5294         cur.id = i;
5295         queries[u].push_back(cur);

```

```

5296     }
5297     vector<ll> ans(q, -1);
5298     SegTree<int> T(n);
5299     const ll inf = 2.5e9;
5300     auto dfs = [&] (const auto &self, int u, int p) -> void {
5301         T.update(b[u], 1);
5302         for (auto qry : queries[u]) {
5303             ll lo = 1, hi = inf;
5304             while (lo < hi) {
5305                 ll mid = (lo + hi)/2;
5306                 ll den = (mid * (mid+1) / 2);
5307                 ll want = (qry.k + den - 1) / den;
5308
5309                 if (want >= n or T.query(want, n) < qry.t) lo = mid+1;
5310                 else hi = mid;
5311             }
5312             if (lo == inf) lo = -1;
5313             ans[qry.id] = lo;
5314         }
5315
5316         for (int v : g[u]) {
5317             if (v == p) continue;
5318             self(self, v, u);
5319         }
5320         T.update(b[u], -1);
5321     };
5322     dfs(dfs, 0, 0);
5323     for (auto x : ans) cout << x << '\n';
5324 }
5325
5326 //ENODE_EASY
5327 #include <map>
5328 #include <set>
5329 #include <cmath>
5330 #include <ctime>
5331 #include <queue>
5332 #include <stack>
5333 #include <cstdio>
5334 #include <cstdlib>
5335 #include <vector>
5336 #include <cstring>
5337 #include <algorithm>
5338 using namespace std;
5339 typedef double db;
5340 typedef long long ll;
5341 typedef unsigned long long ull;
5342 const int N=1000010;
5343 const int LOGN=28;
5344 const ll TMD=0;
5345 const ll INF=2147483647;
5346 int n,q,cur;
5347 int a[N],key[N],tag[N],enough[N],tot[N];
5348 ll ans[N];
5349 vector<int> val;
5350 vector<int> G[N];
5351
5352 struct query
5353 {
5354     int t,id;
5355     ll k;
5356
5357     query() {}
5358
5359     query(int t,ll k,int id):t(t),k(k),id(id) {}
5360 };
5361 vector<query> Q[N];
5362
5363 void DFS(int x,int pre)
5364 {

```

```

5365     tot[x]=tot[pre]+enough[x];
5366     for(int i=0;i<G[x].size();i++)
5367     {
5368         int y=G[x][i];
5369         if(y==pre) continue;
5370         DFS(y,x);
5371     }
5372 }
5373
5374 void init()
5375 {
5376     scanf("%d",&n);
5377     for(int i=1;i<=n;i++) scanf("%d",&a[i]);
5378     for(int i=1;i<n;i++)
5379     {
5380         int u,v;
5381         scanf("%d%d",&u,&v);
5382         G[u].push_back(v);
5383         G[v].push_back(u);
5384     }
5385     scanf("%d",&q);
5386     for(int i=1;i<=q;i++)
5387     {
5388         int x,t;
5389         ll k;
5390         scanf("%d%d%lld",&x,&t,&k);
5391         Q[x].push_back(query(t,k,i));
5392     }
5393 }
5394
5395 void solve()
5396 {
5397     for(int i=1;i<=q;i++) ans[i]=-1;
5398     for(int i=1;i<=n;i++)
5399     {
5400         key[i]+=a[i];
5401         for(int j=0;j<G[i].size();j++) key[i]+=a[G[i][j]];
5402         if(!tag[key[i]]&&key[i]) tag[key[i]]=1,val.push_back(key[i]);
5403     }
5404     sort(val.begin(),val.end(),greater<int>());
5405     for(int i=0;i<val.size();i++)
5406     {
5407         for(int j=1;j<=n;j++) if(key[j]>=val[i]) enough[j]=1;
5408         DFS(1,0);
5409         for(int j=1;j<=n;j++)
5410         {
5411             for(int k=0;k<Q[j].size();k++)
5412             {
5413                 query qu=Q[j][k];
5414                 if(ans[qu.id]==-1&&tot[j]>=qu.t)
5415                 {
5416                     ll L=0,R=(ll)sqrt(qu.k/val[i]*2+1)+2,M;
5417                     while(L+1!=R)
5418                     {
5419                         M=(L+R)>>1;
5420                         if(M*(1+M)/2*val[i]>=qu.k) R=M;
5421                         else L=M;
5422                     }
5423                     ans[qu.id]=R;
5424                 }
5425             }
5426         }
5427     }
5428     for(int i=1;i<=q;i++) printf("%lld\n",ans[i]);
5429 }
5430
5431 int main()
5432 {
5433     init();

```



```

5434     solve();
5435
5436     return 0;
5437 }
5438
5439 //ENODE_EASY-EDITOR
5440 #include "bits/stdc++.h"
5441 // #pragma GCC optimize("O3,unroll-loops")
5442 // #pragma GCC target("avx2,bmi,bmi2,lzcnt,popcnt")
5443 using namespace std;
5444 using ll = long long int;
5445 mt19937_64 rng(chrono::high_resolution_clock::now().time_since_epoch().count());
5446
5447 struct query {
5448     ll t, k, id;
5449 };
5450
5451 /**
5452  * Point-update Segment Tree
5453  * Source: kactl
5454  * Description: Iterative point-update segment tree, ranges are half-open i.e [L, R).
5455  *              f is any associative function.
5456  * Time: O(logn) update/query
5457  */
5458
5459 template<class T, T unit = T()>
5460 struct SegTree {
5461     T f(T a, T b) { return a+b; }
5462     vector<T> s; int n;
5463     SegTree(int _n = 0, T def = unit) : s(2*_n, def), n(_n) {}
5464     void update(int pos, T val) {
5465         for (s[pos += n] += val; pos /= 2;)
5466             s[pos] = f(s[pos * 2], s[pos * 2 + 1]);
5467     }
5468     T query(int b, int e) {
5469         T ra = unit, rb = unit;
5470         for (b += n, e += n; b < e; b /= 2, e /= 2) {
5471             if (b % 2) ra = f(ra, s[b++]);
5472             if (e % 2) rb = f(s[--e], rb);
5473         }
5474         return f(ra, rb);
5475     }
5476 };
5477
5478 int main()
5479 {
5480     ios::sync_with_stdio(false); cin.tie(0);
5481
5482     int n; cin >> n;
5483     vector<int> a(n), b(n);
5484     for (int &x : a) cin >> x;
5485     vector<vector<int>> g(n);
5486     for (int i = 0; i < n-1; ++i) {
5487         int u, v; cin >> u >> v;
5488         g[--u].push_back(--v);
5489         g[v].push_back(u);
5490     }
5491     for (int i = 0; i < n; ++i) {
5492         b[i] = a[i];
5493         for (int u : g[i]) b[i] += a[u];
5494     }
5495     vector<vector<query>> queries(n);
5496     int q; cin >> q;
5497     for (int i = 0; i < q; ++i) {
5498         query cur;
5499         int u; cin >> u; --u;
5500         cin >> cur.t >> cur.k;
5501         cur.id = i;
5502         queries[u].push_back(cur);

```

```

5503     }
5504     vector<ll> ans(q, -1);
5505     SegTree<int> T(n);
5506     const ll inf = 2.5e9;
5507     auto dfs = [&] (const auto &self, int u, int p) -> void {
5508         T.update(b[u], 1);
5509         for (auto qry : queries[u]) {
5510             ll lo = 1, hi = inf;
5511             while (lo < hi) {
5512                 ll mid = (lo + hi)/2;
5513                 ll den = (mid * (mid+1) / 2);
5514                 ll want = (qry.k + den - 1) / den;
5515
5516                 if (want >= n or T.query(want, n) < qry.t) lo = mid+1;
5517                 else hi = mid;
5518             }
5519             if (lo == inf) lo = -1;
5520             ans[qry.id] = lo;
5521         }
5522
5523         for (int v : g[u]) {
5524             if (v == p) continue;
5525             self(self, v, u);
5526         }
5527         T.update(b[u], -1);
5528     };
5529     dfs(dfs, 0, 0);
5530     for (auto x : ans) cout << x << '\n';
5531 }
5532
5533 //DIVSORT
5534 #include "bits/stdc++.h"
5535 // #pragma GCC optimize("O3,unroll-loops")
5536 // #pragma GCC target("avx2,bmi,bmi2,lzcnt,popcnt")
5537 using namespace std;
5538 using ll = long long int;
5539 mt19937_64 rng(chrono::high_resolution_clock::now().time_since_epoch().count());
5540
5541 int main()
5542 {
5543     ios::sync_with_stdio(false); cin.tie(0);
5544
5545     const int MX = 5e5 + 10;
5546     vector<basic_string<int>> facs(MX);
5547     vector<int> val(MX);
5548     for (int i = 1; i < MX; ++i) for (int j = i; j < MX; j += i) {
5549         facs[j].push_back(i);
5550         if (j > i) val[j] = 1 + val[i];
5551     }
5552
5553     int t; cin >> t;
5554     while (t--) {
5555         int n; cin >> n;
5556         vector<int> a(n+1);
5557         a[0] = 1;
5558         for (int i = 1; i <= n; ++i) {
5559             cin >> a[i];
5560         }
5561
5562         const int SZ = 210;
5563         array<int, SZ> dp{};
5564         int ans = 0;
5565         for (int i = 1; i <= n; ++i) {
5566             array<int, SZ> curdp{};
5567             int ptr = 0, mn = INT_MAX;
5568             ans = INT_MAX;
5569             auto &prv = facs[a[i-1]];
5570             auto &cur = facs[a[i]];
5571             for (int j = 0; j < cur.size(); ++j) {

```

```

5572         int d = cur[j];
5573         while (ptr < prv.size()) {
5574             if (prv[ptr] <= d) {
5575                 mn = min(mn, dp[ptr]);
5576                 ++ptr;
5577             }
5578             else break;
5579         }
5580         curdp[j] = mn + val[a[i]/d];
5581         ans = min(ans, curdp[j]);
5582     }
5583     swap(dp, curdp);
5584 }
5585 cout << ans << '\n';
5586 }
5587 }
5588
5589 //PALSWAP
5590 #include "bits/stdc++.h"
5591 // #pragma GCC optimize("O3,unroll-loops")
5592 // #pragma GCC target("avx2,bmi,bmi2,lzcnt,popcnt")
5593 using namespace std;
5594 using ll = long long int;
5595 mt19937_64 rng(chrono::high_resolution_clock::now().time_since_epoch().count());
5596
5597 #include <bits/extc++.h>
5598 using namespace __gnu_pbds;
5599 template<class T>
5600 using Tree = tree<T, null_type, less<T>, rb_tree_tag, tree_order_statistics_node_update>;
5601
5602 int main()
5603 {
5604     ios::sync_with_stdio(false); cin.tie(0);
5605
5606     auto invcount = [&] (const auto &v) {
5607         Tree<int> T;
5608         ll ret = 0;
5609         for (int x : v) {
5610             ret += T.size() - T.order_of_key(x);
5611             T.insert(x);
5612         }
5613         return ret;
5614     };
5615
5616     auto calc = [&] (const string &s, const string &t) {
5617         int n = s.size();
5618         vector<vector<int>> pos(26);
5619         for (int i = 0; i < n; ++i) {
5620             pos[s[i] - 'a'].push_back(i);
5621         }
5622         vector<int> ptr(26), reach(n);
5623         for (int i = 0; i < n; ++i) {
5624             int c = t[i] - 'a';
5625             if (ptr[c] == (int)pos[c].size()) return -1LL; // Not possible
5626             reach[i] = pos[c][ptr[c]];
5627             ++ptr[c];
5628         }
5629         return invcount(reach);
5630     };
5631
5632     int t; cin >> t;
5633     while (t--) {
5634         int n, m; cin >> n >> m;
5635         vector<string> v(n);
5636         for (auto &s : v) cin >> s;
5637         ll ans = 0;
5638         for (int i = 0; i < m-1-i; ++i) {
5639             string a = "", b = "";
5640             for (int j = 0; j < n; ++j) {

```

```

5641         a += v[j][i];
5642         b += v[j][m-1-i];
5643     }
5644     ll res = calc(a, b);
5645     if (res == -1) {
5646         ans = -1;
5647         break;
5648     }
5649     ans += res;
5650 }
5651 cout << ans << '\n';
5652 }
5653 }
5654
5655 //AJ
5656 #include <bits/stdc++.h>
5657 #define ll long long
5658 #define int long long
5659 #define fi first
5660 #define se second
5661 #define mat vector<vector<ll>>
5662 using namespace std;
5663 void db() {cout << endl;}
5664 template <typename T, typename ...U> void db(T a, U ...b) {cout << a << ' ', db(b...);}
5665 #ifdef Cloud
5666 #define file freopen("input.txt", "r", stdin), freopen("output.txt", "w", stdout)
5667 #else
5668 #define file ios::sync_with_stdio(false); cin.tie(0)
5669 #endif
5670 const int N = 2e5 + 1, mod = 1e9 + 7; // inf = 1e9;
5671 const int inf = 1e9;
5672 int d[N], anc[N][20];
5673 vector<int> g[N];
5674 pair<int, int> dfs(int u, int p){
5675     pair<int, int> ans = {0, u};
5676     for (int i : g[u]){
5677         if (i == p) continue;
5678         d[i] = d[u] + 1;
5679         anc[i][0] = u;
5680         for (int j = 1; j < 20; j++) anc[i][j] = anc[anc[i][j - 1]][j - 1];
5681         pair<int, int> tmp = dfs(i, u);
5682         if (tmp.fi + 1 > ans.fi) ans = {tmp.fi + 1, tmp.se};
5683     }
5684     return ans;
5685 }
5686 int lca(int u, int v){
5687     if (d[u] < d[v]) swap(u, v);
5688     for (int i = 19; i >= 0; i--) if (d[anc[u][i]] >= d[v]) u = anc[u][i];
5689     if (u == v) return v;
5690     for (int i = 19; i >= 0; i--) if (anc[u][i] != anc[v][i]) u = anc[u][i], v = anc[v][i];
5691     return anc[u][0];
5692 }
5693 int dis(int u, int v){
5694     return d[u] + d[v] - 2 * d[lca(u, v)];
5695 }
5696 signed main(){
5697     file;
5698     int n, q;
5699     cin >> n >> q;
5700     for (int i = 1; i <= n; i++) g[i].clear();
5701     for (int i = 0; i < n - 1; i++){
5702         int u, v;
5703         cin >> u >> v;
5704         g[u].push_back(v);
5705         g[v].push_back(u);
5706     }
5707     auto x = dfs(1, -1);
5708     d[x.se] = 0;

```

```

5709     auto y = dfs(x.se, -1);
5710     int ans[n + 1]{};
5711     for (int i = 1; i <= n; i++) ans[i] = max(ans[i], d[i]);
5712     d[y.se] = 1;
5713     dfs(y.se, -1);
5714     for (int i = 1; i <= n; i++) ans[i] = max(ans[i], d[i] - 1);
5715     while (q--){
5716         int u, v;
5717         cin >> u >> v;
5718         int res = ans[v], D = dis(u, v);
5719         while ((res * (res + 1) / 2) % 2 != D % 2) res--;
5720         cout << res + 1 << '\n';
5721     }
5722 }
5723
5724 //AJ-EDITOR
5725 #include "bits/stdc++.h"
5726 // #pragma GCC optimize("O3,unroll-loops")
5727 // #pragma GCC target("avx2,bmi,bmi2,lzcnt,popcnt")
5728 using namespace std;
5729 using ll = long long int;
5730 mt19937_64 rng(chrono::high_resolution_clock::now().time_since_epoch().count());
5731
5732 int main()
5733 {
5734     ios::sync_with_stdio(false); cin.tie(0);
5735
5736     int n, m; cin >> n >> m;
5737     vector<vector<int>> g(n);
5738     for (int i = 0; i < n-1; ++i) {
5739         int u, v; cin >> u >> v;
5740         g[--u].push_back(--v);
5741         g[v].push_back(u);
5742     }
5743     auto bfs = [&] (int src) {
5744         vector<int> dist(n, -1);
5745         dist[src] = 0;
5746         queue<int> q; q.push(src);
5747         while (!q.empty()) {
5748             int u = q.front(); q.pop();
5749             for (int v : g[u]) {
5750                 if (dist[v] == -1) {
5751                     dist[v] = 1 + dist[u];
5752                     q.push(v);
5753                 }
5754             }
5755         }
5756         return dist;
5757     };
5758     auto tmp = bfs(0);
5759     int u1 = max_element(begin(tmp), end(tmp)) - begin(tmp);
5760     auto d1 = bfs(u1);
5761     int u2 = max_element(begin(d1), end(d1)) - begin(d1);
5762     auto d2 = bfs(u2);
5763
5764     while (m--) {
5765         int u, v; cin >> u >> v; --u, --v;
5766         int mx = max(d1[v], d2[v]);
5767         int par = (d1[u]%2)^(d1[v]%2);
5768         while (1) {
5769             int par2 = ((1LL*mx*(mx+1))/2)%2;
5770             if (par == par2) break;
5771             --mx;
5772         }
5773         cout << mx+1 << '\n';
5774     }
5775 }
5776
5777 //ORTUPLES

```

```

5778 #include <bits/stdc++.h>
5779 using namespace std;
5780
5781 int main() {
5782     //freopen("inpl.in", "r", stdin);
5783     //freopen("inpl.out", "w", stdout);
5784     int t;
5785     cin >> t;
5786     while(t--) {
5787         int x, y, z;
5788         cin >> x >> y >> z;
5789         long long ans = 1;
5790         for(int i = 0; i < 20; i++) {
5791             int cnt = ((x&(1<<i)) > 0);
5792             cnt += ((y&(1<<i)) > 0);
5793             cnt += ((z&(1<<i)) > 0);
5794             if(cnt == 1) ans = 0;
5795             else if(cnt == 3) ans *= 4;
5796         }
5797         cout << ans << "\n";
5798     }
5799 }
5800
5801 //NZXOR
5802 #ifdef WTSH
5803     #include <wtsh.h>
5804 #else
5805     #include <bits/stdc++.h>
5806     using namespace std;
5807     #define dbg(...)
5808 #endif
5809
5810 #define int long long
5811 #define endl "\n"
5812 #define sz(w) (int)(w.size())
5813 using pii = pair<int, int>;
5814
5815 const long long INF = 1e18;
5816
5817 const int N = 1e6 + 5;
5818
5819 // ----- Input Checker Start -----
5820
5821 long long readInt(long long l, long long r, char endd)
5822 {
5823     long long x = 0;
5824     int cnt = 0, fi = -1;
5825     bool is_neg = false;
5826     while(true)
5827     {
5828         char g = getchar();
5829         if(g == '-')
5830         {
5831             assert(fi == -1);
5832             is_neg = true;
5833             continue;
5834         }
5835         if('0' <= g && g <= '9')
5836         {
5837             x *= 10;
5838             x += g - '0';
5839             if(cnt == 0)
5840                 fi = g - '0';
5841             cnt++;
5842             assert(fi != 0 || cnt == 1);
5843             assert(fi != 0 || is_neg == false);
5844             assert(!(cnt > 19 || (cnt == 19 && fi > 1)));
5845         }
5846         else if(g == endd)

```

```

5847     {
5848         if(is_neg)
5849             x = -x;
5850         if(!(l <= x && x <= r))
5851         {
5852             cerr << "L: " << l << ", R: " << r << ", Value Found: " << x << '\n';
5853             assert(false);
5854         }
5855         return x;
5856     }
5857     else
5858     {
5859         assert(false);
5860     }
5861 }
5862 }
5863
5864 string readString(int l, int r, char endd)
5865 {
5866     string ret = "";
5867     int cnt = 0;
5868     while(true)
5869     {
5870         char g = getchar();
5871         assert(g != -1);
5872         if(g == endd)
5873             break;
5874         cnt++;
5875         ret += g;
5876     }
5877     assert(l <= cnt && cnt <= r);
5878     return ret;
5879 }
5880
5881 long long readIntSp(long long l, long long r) { return readInt(l, r, ' '); }
5882 long long readIntLn(long long l, long long r) { return readInt(l, r, '\n'); }
5883 string readStringSp(int l, int r) { return readString(l, r, ' '); }
5884 string readStringLn(int l, int r) { return readString(l, r, '\n'); }
5885 void readEOF() { assert(getchar() == EOF); }
5886
5887 vector<int> readVectorInt(int n, long long l, long long r)
5888 {
5889     vector<int> a(n);
5890     for(int i = 0; i < n - 1; i++)
5891         a[i] = readIntSp(l, r);
5892     a[n - 1] = readIntLn(l, r);
5893     return a;
5894 }
5895
5896 // ----- Input Checker End -----
5897
5898 int sumN = 0;
5899
5900 void solve()
5901 {
5902     int n = readIntLn(1, 1e5);
5903     sumN += n;
5904     vector<int> a = readVectorInt(n, 0, (1 << 30) - 1);
5905     set<int> prefxor{0};
5906     int ans = 0, cur = 0;
5907     for(int &x: a)
5908     {
5909         cur ^= x;
5910         if(prefxor.count(cur))
5911         {
5912             ans++;
5913             prefxor.clear();
5914             cur = 0;
5915             prefxor.insert(0);

```

```

5916         }
5917     else
5918     {
5919         prefixor.insert(cur);
5920     }
5921 }
5922 cout << ans << endl;
5923 }
5924
5925 int32_t main()
5926 {
5927     ios::sync_with_stdio(0);
5928     cin.tie(0);
5929     int T = readIntLn(1, 1e5);
5930     for(int tc = 1; tc <= T; tc++)
5931     {
5932         // cout << "Case #" << tc << ": ";
5933         solve();
5934     }
5935     readEOF();
5936     assert(sumN <= 3e5);
5937     return 0;
5938 }
5939
5940 //BALLSEQ
5941 #include <iostream>
5942 #include <iomanip>
5943 #include <fstream>
5944 #include <vector>
5945 #include <array>
5946 #include <algorithm>
5947 #include <utility>
5948 #include <map>
5949 #include <queue>
5950 #include <set>
5951 #include <cmath>
5952 #include <cstdio>
5953 #include <cstring>
5954 #include <functional>
5955 #include <numeric>
5956
5957 #define ll long long
5958 #define ld long double
5959 #define eps 1e-8
5960 #define MOD 1000000007
5961
5962 #define INF 0x3f3f3f3f
5963 #define INFL 0x3f3f3f3f3f3f3f3f
5964
5965 // change if necessary
5966 #define MAXN 1000000
5967
5968 using namespace std;
5969
5970 void solve() {
5971     int n, k;
5972     cin >> n >> k;
5973
5974     int cur_val = 0;
5975     int prev = 0;
5976
5977     for (int i = 0; i < k; i++) {
5978         int a;
5979         cin >> a;
5980         cur_val += a - prev - 1;
5981         cur_val >= 1;
5982         prev = a;
5983     }
5984

```



```

5985     cur_val += n - prev;
5986
5987     cout << (n - k) - __builtin_popcount(cur_val) << '\n';
5988 }
5989
5990 int main() {
5991     cin.tie(0)->sync_with_stdio(0);
5992
5993     int t;
5994     cin >> t;
5995
5996     while (t--) {
5997         solve();
5998     }
5999
6000     return 0;
6001 }
6002
6003 //DISGUST
6004 #include <bits/stdc++.h>
6005 using namespace std;
6006
6007 void solve_case();
6008
6009 const long long inf = 1e16;
6010 signed main() {
6011     std::ios::sync_with_stdio(0);
6012     std::cout.tie(0);
6013     std::cin.tie(0);
6014     int tt = 1;
6015     std::cin >> tt;
6016     while(tt--) {
6017         solve_case();
6018     }
6019     return 0;
6020 }
6021
6022 void solve_case() {
6023     int n, m;
6024     cin >> n >> m;
6025     vector<vector<int>>> a(n, vector<int>(n));
6026     vector<vector<long long>>> dt(n, vector<long long>(n, inf)), f(n, vector<long
long>(n, inf));
6027     for(int i = 0; i < n; i++) {
6028         dt[i][i] = 0;
6029         for(int j = 0; j < n; j++)
6030             cin >> a[i][j], a[i][j]--;
6031     }
6032     for(int x, y, z, i = 0; i < m; i++) {
6033         cin >> x >> y >> z; --x, --y;
6034         dt[x][y] = min(dt[x][y], (long long) z);
6035     }
6036     for(int j = 0; j < n; j++)
6037         for(int i = 0; i < n; i++)
6038             for(int k = 0; k < n; k++)
6039                 dt[i][k] = min(dt[i][k], dt[i][j] + dt[j][k]);
6040
6041     for(int i = 0; i < n; i++)
6042         for(int j = 0; j < n; j++)
6043             for(int k = 0; k < n; k++)
6044                 f[i][j] = min(f[i][j], dt[i][k] + 2 * (k - j) * (k - j)); //twice to
unify cost/disgust
6045
6046     long long ans = 0;
6047     for(int i = 0; i < n; i++)
6048         for(int j = i; j < n; j++) {
6049             long long best = inf;
6050             for(int k = 0; k < n; k++)
6051                 best = min({best, f[a[i][j]][k] + dt[a[j][i]][k], f[a[j][i]][k] +

```

```

        dt[a[i][j]][k]));
6052     ans += best;
6053     //cout << i << ' ' << j << ' ' << a[i][j] + 1 << ' ' << a[j][i] + 1 << '\n';
6054     //cout << "best: " << best << '\n';
6055 }
6056 cout << ans << '\n';
6057 }
6058
6059 //DISGUST-EDITOR - FLOYD WARSHAL
6060 #include "bits/stdc++.h"
6061 // #pragma GCC optimize("O3,unroll-loops")
6062 // #pragma GCC target("avx2,bmi,bmi2,lzcnt,popcnt")
6063 using namespace std;
6064 using ll = long long int;
6065 mt19937_64 rng(chrono::high_resolution_clock::now().time_since_epoch().count());
6066
6067 // CHT; query returns max of lines in set
6068 struct Line {
6069     mutable ll k, m, p;
6070     bool operator<(const Line& o) const { return k < o.k; }
6071     bool operator<(ll x) const { return p < x; }
6072 };
6073 struct LineContainer : multiset<Line, less<>> {
6074     // (for doubles, use inf = 1/.0, div(a,b) = a/b)
6075     const ll inf = LLONG_MAX;
6076     ll div(ll a, ll b) { // floored division
6077         return a / b - ((a ^ b) < 0 && a % b); }
6078     bool isect(iterator x, iterator y) {
6079         if (y == end()) { x->p = inf; return false; }
6080         if (x->k == y->k) x->p = x->m > y->m ? inf : -inf;
6081         else x->p = div(y->m - x->m, x->k - y->k);
6082         return x->p >= y->p;
6083     }
6084     void add(ll k, ll m) {
6085         auto z = insert({k, m, 0}), y = z++, x = y;
6086         while (isect(y, z)) z = erase(z);
6087         if (x != begin() && isect(--x, y)) isect(x, y = erase(y));
6088         while ((y = x) != begin() && (--x)->p >= y->p)
6089             isect(x, erase(y));
6090     }
6091     ll query(ll x) {
6092         assert(!empty());
6093         auto l = *lower_bound(x);
6094         return l.k * x + l.m;
6095     }
6096 };
6097
6098 int main()
6099 {
6100     ios::sync_with_stdio(false); cin.tie(0);
6101
6102     const ll inf = 1e18 + 5;
6103     int t; cin >> t;
6104     while (t--) {
6105         int n, m; cin >> n >> m;
6106         vector a(n*n, 0);
6107         for (int &x : a) cin >> x;
6108         vector dist(n+1, vector(n+1, inf));
6109         for (int i = 1; i <= n; ++i) dist[i][i] = 0;
6110         for (int i = 0; i < m; ++i) {
6111             ll x, y, z; cin >> x >> y >> z;
6112             dist[x][y] = min(dist[x][y], z);
6113         }
6114         for (int k = 0; k < n; ++k) {
6115             for (int i = 1; i <= n; ++i) {
6116                 for (int j = 1; j <= n; ++j) {
6117                     dist[i][j] = min(dist[i][j], dist[i][k] + dist[k][j]);
6118                 }
6119             }

```

```

6120     }
6121
6122     vector<LineContainer> LC(n+1);
6123     for (int i = 1; i <= n; ++i) {
6124         for (int j = 1; j <= n; ++j) {
6125             if (dist[i][j] > inf/2) continue;
6126             LC[i].add(4*j, -dist[i][j] - 2*j*j);
6127         }
6128     }
6129
6130     ll ans = 0;
6131     for (int i = 0; i < n; ++i) {
6132         for (int j = i+1; j < n; ++j) {
6133             int x = a[i*n + j], y = a[i + j*n];
6134             ll cost = inf;
6135             for (int k = 1; k <= n; ++k) {
6136                 // end at (k, z) with cost:
6137                 //  $2k^2 + 2z^2 - 4*k*z + dist[x][k] + dist[y][z]$ 
6138                 //  $(2k^2 + dist[x][k]) + \min(dist[y][z] + 2z^2 - 4*k*z)$  across all z
6139                 cost = min(cost, 2*k*k + dist[x][k] - LC[y].query(k));
6140             }
6141             ans += cost;
6142         }
6143     }
6144     cout << ans << '\n';
6145 }
6146 }
6147
6148 //EQDIS
6149 #include<bits/stdc++.h>
6150 using namespace std;
6151
6152 int main() {
6153     int tt;
6154     cin >> tt;
6155     while (tt--) {
6156         int n;
6157         cin >> n;
6158         vector<int> a(n);
6159         for (int i = 1; i <= n; i++) {
6160             int x;
6161             cin >> x;
6162             a[x - 1]++;
6163         }
6164         if (n % 2 == 1 && *max_element(a.begin(), a.end()) == 1) {
6165             cout << "NO\n";
6166         } else {
6167             cout << "YES\n";
6168         }
6169     }
6170     return 0;
6171 }
6172
6173 //DPRTMNTS-EASY
6174 /**
6175  * the_hyp0cr1t3
6176  * 30.08.2022 01:23:14
6177  */
6178 #ifdef W
6179     #include <k_II.h>
6180 #else
6181     #include <bits/stdc++.h>
6182     using namespace std;
6183 #endif
6184
6185 int main() {
6186     #if __cplusplus > 201703L
6187         namespace R = ranges;
6188     #endif

```

```

6189 ios_base::sync_with_stdio(false), cin.tie(nullptr);
6190
6191 int tests; cin >> tests;
6192 while(tests--) [&] {
6193     int i, N, K;
6194     cin >> N >> K;
6195
6196     if(K == N * (N - 1) / 2) {
6197         // There is just one department.
6198         while(K--) cin >> i >> i;
6199
6200         cout << 1 << '\n';
6201         for(i = 0; i < N; i++)
6202             cout << 1 << " \n"[i + 1 == N];
6203         cout << 1 << '\n';
6204
6205     } else {
6206
6207         vector<int> deg(N);
6208         vector<vector<int>> adj(N);
6209         while(K--) {
6210             int u, v; cin >> u >> v;
6211             ++deg[--u];
6212             ++deg[--v];
6213             adj[u].push_back(v);
6214             adj[v].push_back(u);
6215         }
6216
6217         // Find the employee with minimum degree (guaranteed not to be a leader).
6218         int min_v = min_element(deg.begin(), deg.end()) - deg.begin();
6219
6220         int leader;
6221         vector<bool> mark(N, false);
6222         mark[min_v] = true;
6223         for(int u: adj[min_v]) {
6224             if(deg[u] == deg[min_v]) {
6225                 mark[u] = true;
6226             } else {
6227                 // The department head is the sole vertex connected
6228                 // to [min_v] with a greater degree.
6229                 leader = u;
6230             }
6231         }
6232
6233         // Having marked all the vertices in the department of [min_v],
6234         // the unmarked vertices connected to [leader] are all the leaders of all
6235         // the other departments.
6236         // Getting their department sizes is easy and can be done as follows.
6237         int M = 0;
6238         vector<int> col(N, -1), leaders;
6239         adj[leader].push_back(leader);
6240         for(int u: adj[leader]) {
6241             if(!mark[u]) {
6242                 col[u] = M++;
6243                 leaders.push_back(u);
6244             }
6245         }
6246
6247         for(int v: leaders)
6248             for(int u: adj[v])
6249                 if(col[u] == -1) col[u] = col[v];
6250
6251         cout << M << '\n';
6252         for(i = 0; i < N; i++)
6253             cout << col[i] + 1 << " \n"[i + 1 == N];
6254         for(i = 0; i < M; i++)
6255             cout << leaders[i] + 1 << " \n"[i + 1 == M];
6256     }

```

```

6257     }();
6258
6259 } // ~W
6260
6261 //DPRTMNTS-HARD
6262 /**
6263  * the_hyp0cr1t3
6264  * 30.08.2022 01:23:14
6265  */
6266 #ifdef W
6267     #include <k_II.h>
6268 #else
6269     #include <bits/stdc++.h>
6270     using namespace std;
6271 #endif
6272
6273 int main() {
6274     #if __cplusplus > 201703L
6275         namespace R = ranges;
6276     #endif
6277     ios_base::sync_with_stdio(false), cin.tie(nullptr);
6278
6279     int tests; cin >> tests;
6280     while(tests--) [&] {
6281         int i, N, K;
6282         cin >> N >> K;
6283
6284         vector<int> deg(N);
6285         vector<vector<int>> adj(N);
6286         while(K--) {
6287             int u, v; cin >> u >> v;
6288             ++deg[--u]; ++deg[--v];
6289             adj[u].push_back(v);
6290             adj[v].push_back(u);
6291         }
6292
6293         int min_v = min_element(deg.begin(), deg.end()) - deg.begin();
6294
6295         vector<int> leaders, col(N, -1);
6296         vector<bool> mark(N, false);
6297         mark[min_v] = true;
6298         for(int u: adj[min_v]) {
6299             if(deg[u] == deg[min_v])
6300                 mark[u] = true;
6301             else
6302                 leaders.push_back(u);
6303         }
6304
6305         if(leaders.size() != 1) {
6306             leaders = adj[min_v];
6307             leaders.push_back(min_v);
6308         } else {
6309             for(int u: adj[leaders[0]])
6310                 if(!mark[u]) leaders.push_back(u);
6311         }
6312
6313         int M = 0;
6314         for(int l: leaders) col[l] = M++;
6315
6316         for(int l: leaders)
6317             for(int u: adj[l])
6318                 if(col[u] == -1) col[u] = col[l];
6319
6320         cout << M << '\n';
6321         for(i = 0; i < N; i++)
6322             cout << col[i] + 1 << " \n"[i + 1 == N];
6323         for(i = 0; i < M; i++)
6324             cout << leaders[i] + 1 << " \n"[i + 1 == M];
6325     }();

```

```

6326
6327 } // ~W
6328
6329 //DPRTMNTS-EDITOR
6330 #include "bits/stdc++.h"
6331 // #pragma GCC optimize("O3,unroll-loops")
6332 // #pragma GCC target("avx2,bmi,bmi2,lzcnt,popcnt")
6333 using namespace std;
6334 using ll = long long int;
6335 mt19937_64 rng(chrono::high_resolution_clock::now().time_since_epoch().count());
6336
6337 int main()
6338 {
6339     ios::sync_with_stdio(false); cin.tie(0);
6340
6341     int t; cin >> t;
6342     while (t--) {
6343         int n, k; cin >> n >> k;
6344         vector adj(n+1, vector(n+1, 0));
6345         for (int i = 0; i < k; ++i) {
6346             int u, v; cin >> u >> v;
6347             adj[u][v] = adj[v][u] = 1;
6348         }
6349         int ch = 0, mx = 0;
6350         for (int i = 1; i <= n; ++i) {
6351             int deg = count(begin(adj[i]), end(adj[i]), 1);
6352             if (deg > mx) {
6353                 mx = deg;
6354                 ch = i;
6355             }
6356         }
6357         vector<int> cand1, cand2;
6358         for (int i = 1; i <= n; ++i) {
6359             if (!adj[ch][i]) continue;
6360             if (cand1.empty()) cand1.push_back(i);
6361             else {
6362                 if (adj[cand1[0]][i]) cand1.push_back(i);
6363                 else cand2.push_back(i);
6364             }
6365         }
6366         cand1.push_back(ch); cand2.push_back(ch);
6367         auto check = [&] (auto cand) {
6368             vector<int> mark(n+1);
6369             int idx = 1;
6370             for (int x : cand) mark[x] = idx++;
6371             for (int i = 1; i <= n; ++i) {
6372                 if (mark[i]) continue;
6373                 int found = 0;
6374                 for (int x : cand) {
6375                     if (adj[i][x]) {
6376                         mark[i] = mark[x];
6377                         ++found;
6378                     }
6379                 }
6380                 if (found != 1) return false;
6381             }
6382             cout << size(cand) << '\n';
6383             for (int i = 1; i <= n; ++i) cout << mark[i] << ' ';
6384             cout << '\n';
6385             for (int x : cand) cout << x << ' ';
6386             cout << '\n';
6387             return true;
6388         };
6389         assert(check(cand1) || check(cand2));
6390     }
6391 }
6392
6393
6394

```

```

6395 //SUBARRAYGAME
6396 #ifndef WTSH
6397     #include <wtsh.h>
6398 #else
6399     #include <bits/stdc++.h>
6400     using namespace std;
6401     #define dbg(...)
6402 #endif
6403
6404 #define int long long
6405 #define endl "\n"
6406 #define sz(w) (int)(w.size())
6407 using pii = pair<int, int>;
6408
6409 const long long INF = 1e18;
6410
6411 const int N = 1e6 + 5;
6412
6413 // ----- Input Checker Start -----
6414
6415 long long readInt(long long l, long long r, char endd)
6416 {
6417     long long x = 0;
6418     int cnt = 0, fi = -1;
6419     bool is_neg = false;
6420     while(true)
6421     {
6422         char g = getchar();
6423         if(g == '-')
6424         {
6425             assert(fi == -1);
6426             is_neg = true;
6427             continue;
6428         }
6429         if('0' <= g && g <= '9')
6430         {
6431             x *= 10;
6432             x += g - '0';
6433             if(cnt == 0)
6434                 fi = g - '0';
6435             cnt++;
6436             assert(fi != 0 || cnt == 1);
6437             assert(fi != 0 || is_neg == false);
6438             assert(!(cnt > 19 || (cnt == 19 && fi > 1)));
6439         }
6440         else if(g == endd)
6441         {
6442             if(is_neg)
6443                 x = -x;
6444             if(!(l <= x && x <= r))
6445             {
6446                 cerr << "L: " << l << ", R: " << r << ", Value Found: " << x << '\n';
6447                 assert(false);
6448             }
6449             return x;
6450         }
6451         else
6452         {
6453             assert(false);
6454         }
6455     }
6456 }
6457
6458 string readString(int l, int r, char endd)
6459 {
6460     string ret = "";
6461     int cnt = 0;
6462     while(true)
6463     {

```

```

6464         char g = getchar();
6465         assert(g != -1);
6466         if(g == endd)
6467             break;
6468         cnt++;
6469         ret += g;
6470     }
6471     assert(l <= cnt && cnt <= r);
6472     return ret;
6473 }
6474
6475 long long readIntSp(long long l, long long r) { return readInt(l, r, ' '); }
6476 long long readIntLn(long long l, long long r) { return readInt(l, r, '\n'); }
6477 string readStringSp(int l, int r) { return readString(l, r, ' '); }
6478 string readStringLn(int l, int r) { return readString(l, r, '\n'); }
6479 void readEOF() { assert(getchar() == EOF); }
6480
6481 vector<int> readVectorInt(int n, long long l, long long r)
6482 {
6483     vector<int> a(n);
6484     for(int i = 0; i < n - 1; i++)
6485         a[i] = readIntSp(l, r);
6486     a[n - 1] = readIntLn(l, r);
6487     return a;
6488 }
6489
6490 // ----- Input Checker End -----
6491
6492 int sumN = 0, Alice = 0, Bob = 0;
6493
6494 void solve()
6495 {
6496     int n = readIntLn(1, 1e5);
6497     sumN += n;
6498     vector<int> a = readVectorInt(n, 1, 1e9);
6499     assert(sz(set<int>(a.begin(), a.end())) == n);
6500     int ans = 0;
6501     for(int i = 0; i + 1 < n; )
6502     {
6503         int cnt = 0;
6504         if(a[i] < a[i + 1])
6505         {
6506             while(i + 1 < n and a[i] < a[i + 1])
6507                 i++, cnt++;
6508         }
6509         else
6510         {
6511             while(i + 1 < n and a[i] > a[i + 1])
6512                 i++, cnt++;
6513         }
6514         ans ^= (cnt - 1);
6515     }
6516     if(ans == 0)
6517         cout << "Bob\n", Bob++;
6518     else
6519         cout << "Alice\n", Alice++;
6520 }
6521
6522 int32_t main()
6523 {
6524     ios::sync_with_stdio(0);
6525     cin.tie(0);
6526     int T = readIntLn(1, 1e5);
6527     for(int tc = 1; tc <= T; tc++)
6528     {
6529         solve();
6530     }
6531     readEOF();
6532     assert(sumN <= 5e5);

```



```

6533     cerr << "Alice: " << Alice << endl;
6534     cerr << "Bob: " << Bob << endl;
6535     return 0;
6536 }
6537
6538 //DPRTMNTSEASY
6539 /**
6540  * the_hyp0cr1t3
6541  * 30.08.2022 01:23:14
6542  */
6543 #ifdef W
6544     #include <k_II.h>
6545 #else
6546     #include <bits/stdc++.h>
6547     using namespace std;
6548 #endif
6549
6550 int main() {
6551     #if __cplusplus > 201703L
6552         namespace R = ranges;
6553     #endif
6554     ios_base::sync_with_stdio(false), cin.tie(nullptr);
6555
6556     int tests; cin >> tests;
6557     while(tests--) [&] {
6558         int i, N, K;
6559         cin >> N >> K;
6560
6561         if(K == N * (N - 1) / 2) {
6562             // There is just one department.
6563             while(K--) cin >> i >> i;
6564
6565             cout << 1 << '\n';
6566             for(i = 0; i < N; i++)
6567                 cout << 1 << " \n"[i + 1 == N];
6568             cout << 1 << '\n';
6569
6570         } else {
6571
6572             vector<int> deg(N);
6573             vector<vector<int>> adj(N);
6574             while(K--) {
6575                 int u, v; cin >> u >> v;
6576                 ++deg[--u];
6577                 ++deg[--v];
6578                 adj[u].push_back(v);
6579                 adj[v].push_back(u);
6580             }
6581
6582             // Find the employee with minimum degree (guaranteed not to be a leader).
6583             int min_v = min_element(deg.begin(), deg.end()) - deg.begin();
6584
6585             int leader;
6586             vector<bool> mark(N, false);
6587             mark[min_v] = true;
6588             for(int u: adj[min_v]) {
6589                 if(deg[u] == deg[min_v]) {
6590                     mark[u] = true;
6591                 } else {
6592                     // The department head is the sole vertex connected
6593                     // to [min_v] with a greater degree.
6594                     leader = u;
6595                 }
6596             }
6597
6598             // Having marked all the vertices in the department of [min_v],
6599             // the unmarked vertices connected to [leader] are all the leaders of all
6600             // the other departments.
6601             // Getting their department sizes is easy and can be done as follows.

```

```

6601         int M = 0;
6602         vector<int> col(N, -1), leaders;
6603         adj[leader].push_back(leader);
6604         for(int u: adj[leader]) {
6605             if(!mark[u]) {
6606                 col[u] = M++;
6607                 leaders.push_back(u);
6608             }
6609         }
6610
6611         for(int v: leaders)
6612             for(int u: adj[v])
6613                 if(col[u] == -1) col[u] = col[v];
6614
6615         cout << M << '\n';
6616         for(i = 0; i < N; i++)
6617             cout << col[i] + 1 << " \n"[i + 1 == N];
6618         for(i = 0; i < M; i++)
6619             cout << leaders[i] + 1 << " \n"[i + 1 == M];
6620     }
6621
6622     }();
6623
6624 } // ~W
6625
6626 //DPRTMNTSEASY-HARD
6627
6628 /**
6629  * the_hyp0crlt3
6630  * 30.08.2022 01:23:14
6631  */
6632 #ifdef W
6633     #include <k_II.h>
6634 #else
6635     #include <bits/stdc++.h>
6636     using namespace std;
6637 #endif
6638
6639 int main() {
6640     #if __cplusplus > 201703L
6641         namespace R = ranges;
6642     #endif
6643     ios_base::sync_with_stdio(false), cin.tie(nullptr);
6644
6645     int tests; cin >> tests;
6646     while(tests--) [&] {
6647         int i, N, K;
6648         cin >> N >> K;
6649
6650         vector<int> deg(N);
6651         vector<vector<int>> adj(N);
6652         while(K--) {
6653             int u, v; cin >> u >> v;
6654             ++deg[--u]; ++deg[--v];
6655             adj[u].push_back(v);
6656             adj[v].push_back(u);
6657         }
6658
6659         int min_v = min_element(deg.begin(), deg.end()) - deg.begin();
6660
6661         vector<int> leaders, col(N, -1);
6662         vector<bool> mark(N, false);
6663         mark[min_v] = true;
6664         for(int u: adj[min_v]) {
6665             if(deg[u] == deg[min_v])
6666                 mark[u] = true;
6667             else
6668                 leaders.push_back(u);
6669         }

```

```

6670
6671     if(leaders.size() != 1) {
6672         leaders = adj[min_v];
6673         leaders.push_back(min_v);
6674     } else {
6675         for(int u: adj[leaders[0]])
6676             if(!mark[u]) leaders.push_back(u);
6677     }
6678
6679     int M = 0;
6680     for(int l: leaders) col[l] = M++;
6681
6682     for(int l: leaders)
6683         for(int u: adj[l])
6684             if(col[u] == -1) col[u] = col[l];
6685
6686     cout << M << '\n';
6687     for(i = 0; i < N; i++)
6688         cout << col[i] + 1 << " \n"[i + 1 == N];
6689     for(i = 0; i < M; i++)
6690         cout << leaders[i] + 1 << " \n"[i + 1 == M];
6691     }();
6692
6693 } // ~W
6694
6695 //DPRTMNTSEASY-EDITOR
6696 #include "bits/stdc++.h"
6697 // #pragma GCC optimize("O3,unroll-loops")
6698 // #pragma GCC target("avx2,bmi,bmi2,lzcnt,popcnt")
6699 using namespace std;
6700 using ll = long long int;
6701 mt19937_64 rng(chrono::high_resolution_clock::now().time_since_epoch().count());
6702
6703 int main()
6704 {
6705     ios::sync_with_stdio(false); cin.tie(0);
6706
6707     int t; cin >> t;
6708     while (t--) {
6709         int n, k; cin >> n >> k;
6710         vector adj(n+1, vector(n+1, 0));
6711         for (int i = 0; i < k; ++i) {
6712             int u, v; cin >> u >> v;
6713             adj[u][v] = adj[v][u] = 1;
6714         }
6715         int ch = 0, mx = 0;
6716         for (int i = 1; i <= n; ++i) {
6717             int deg = count(begin(adj[i]), end(adj[i]), 1);
6718             if (deg > mx) {
6719                 mx = deg;
6720                 ch = i;
6721             }
6722         }
6723         vector<int> cand1, cand2;
6724         for (int i = 1; i <= n; ++i) {
6725             if (!adj[ch][i]) continue;
6726             if (cand1.empty()) cand1.push_back(i);
6727             else {
6728                 if (adj[cand1[0]][i]) cand1.push_back(i);
6729                 else cand2.push_back(i);
6730             }
6731         }
6732         cand1.push_back(ch); cand2.push_back(ch);
6733         auto check = [&] (auto cand) {
6734             vector<int> mark(n+1);
6735             int idx = 1;
6736             for (int x : cand) mark[x] = idx++;
6737             for (int i = 1; i <= n; ++i) {
6738                 if (mark[i]) continue;

```

```

6739         int found = 0;
6740         for (int x : cand) {
6741             if (adj[i][x]) {
6742                 mark[i] = mark[x];
6743                 ++found;
6744             }
6745         }
6746         if (found != 1) return false;
6747     }
6748     cout << size(cand) << '\n';
6749     for (int i = 1; i <= n; ++i) cout << mark[i] << ' ';
6750     cout << '\n';
6751     for (int x : cand) cout << x << ' ';
6752     cout << '\n';
6753     return true;
6754 };
6755 assert(check(cand1) || check(cand2));
6756 }
6757 }
6758
6759 //RING_GAME
6760 #include <cstdio>
6761 #include <algorithm>
6762 using namespace std;
6763 const int N=100010;
6764 int T,n;
6765 int c[N],cnt[2];
6766
6767 int main()
6768 {
6769     scanf("%d",&T);
6770     while(T--)
6771     {
6772         scanf("%d",&n);
6773         for(int i=1;i<=n;i++) scanf("%d",&c[i]);
6774         cnt[0]=cnt[1]=0;
6775         if(c[1]==c[n]) cnt[c[1]]++;
6776         for(int i=1;i<n;i++) if(c[i]==c[i+1]) cnt[c[i]]++;
6777         printf("%s\n", (min(cnt[0],cnt[1])&1)?"Alice":"Bob");
6778     }
6779
6780     return 0;
6781 }
6782
6783 //WNTR
6784 /*
6785 Yaswanth Phani Kommineni
6786 */
6787 #include <bits/stdc++.h>
6788 using namespace std;
6789 typedef long long int ll;
6790
6791 void solve(){
6792     int n,s = 0;
6793     cin >> n;
6794     vector <int> a(n),b(n);
6795     for(int i=0;i<n;i++){
6796         cin >> a[i];
6797         s += a[i];
6798     }
6799     for(int j=0;j<n;j++){
6800         cin >> b[j];
6801     }
6802     vector < vector < int> > dp(n+1, vector <int> (s+1,-1e9));
6803     dp[0][0] = 0;
6804     for(int i=0;i<n;i++){
6805         for(int j=n;j>=1;j--){
6806             for(int k=s;k>=min(a[i],b[i]);k--){
6807                 dp[j][k] = max(dp[j][k],dp[j-1][k-min(a[i],b[i])] +

```

```

max(0,b[i]-a[i]));
6808     }
6809     }
6810 }
6811 for(int i=n;i>=1;i--){
6812     int cur;
6813     bool fl = false;
6814     for(int j=0;j<=s;j++){
6815         if(s-j <= dp[i][j]){
6816             if(!fl){
6817                 fl = true;
6818                 cur = 2*(s-j);
6819                 continue;
6820             }
6821             cur = min(cur, 2*(s-j));
6822         }
6823     }
6824     if(!fl) cur = -1;
6825     cout << cur << " ";
6826 }
6827 cout << endl;
6828 }
6829
6830 int main(){
6831 #ifndef ONLINE_JUDGE
6832     //freopen("input0.in","r",stdin);
6833     //freopen("input0.out","w",stdout);
6834 #endif
6835     ios_base::sync_with_stdio(false);
6836     cin.tie(0);
6837     cout.tie(0);
6838     int tc;
6839     tc = 1;
6840     cin >> tc;
6841     for(int i=1;i<=tc;i++){
6842         //cout << "Case #" << i << ": ";
6843         solve();
6844     }
6845     return 0;
6846 }
6847
6848 //WNTR-EDITOR
6849 #include "bits/stdc++.h"
6850 // #pragma GCC optimize("O3,unroll-loops")
6851 // #pragma GCC target("avx2,bmi,bmi2,lzcnt,popcnt")
6852 using namespace std;
6853 using ll = long long int;
6854 mt19937_64 rng(chrono::high_resolution_clock::now().time_since_epoch().count());
6855
6856 int main()
6857 {
6858     ios::sync_with_stdio(false); cin.tie(0);
6859
6860     int t; cin >> t;
6861     while (t--) {
6862         int n; cin >> n;
6863         vector<int> a(n), b(n);
6864         for (int &x : a) cin >> x;
6865         for (int &x : b) cin >> x;
6866
6867         int sumA = accumulate(begin(a), end(a), 0);
6868         int sumB = accumulate(begin(b), end(b), 0);
6869         const int inf = 1e9 + 7;
6870         vector dp(n+1, vector(sumB + 1, inf));
6871         vector<int> ans(n, inf);
6872         dp[0][0] = 0;
6873         for (int i = 0; i < n; ++i) {
6874             for (int ch = i+1; ch >= 1; --ch) {
6875                 for (int j = b[i]; j <= sumB; ++j) {

```

```

6876         dp[ch][j] = min(dp[ch][j], dp[ch-1][j-b[i]] + max(0, a[i] - b[i]) -
6877         a[i]);
6878         if (j >= sumA) ans[n-ch] = min(ans[n-ch], 2*dp[ch][j]);
6879     }
6880 }
6881 for (int i = 0; i < n; ++i) {
6882     if (ans[i] < 1e8) cout << ans[i] + 2*sumA << ' ';
6883     else cout << -1 << ' ';
6884 }
6885 cout << '\n';
6886 }
6887 }
6888
6889 //MAKEABEQUAL
6890 //Utkarsh.25dec
6891 #include <bits/stdc++.h>
6892 #define ll long long int
6893 #define pb push_back
6894 #define mp make_pair
6895 #define mod 1000000007
6896 #define vl vector <ll>
6897 #define all(c) (c).begin(), (c).end()
6898 using namespace std;
6899 ll power(ll a, ll b) {ll res=1; a%=mod; assert(b>=0);
6900 for(; b>=1; b--){if(b&1) res=res*a%mod; a=a*a%mod;} return res;}
6901 ll modInverse(ll a){return power(a, mod-2);}
6902 const int N=100023;
6903 bool vis[N];
6904 vector <int> adj[N];
6905 long long readInt(long long l, long long r, char endd){
6906     long long x=0;
6907     int cnt=0;
6908     int fi=-1;
6909     bool is_neg=false;
6910     while(true){
6911         char g=getchar();
6912         if(g=='-'){
6913             assert(fi==-1);
6914             is_neg=true;
6915             continue;
6916         }
6917         if('0'<=g && g<='9'){
6918             x*=10;
6919             x+=g-'0';
6920             if(cnt==0){
6921                 fi=g-'0';
6922             }
6923             cnt++;
6924             assert(fi!=0 || cnt==1);
6925             assert(fi!=0 || is_neg==false);
6926             assert(!(cnt>19 || ( cnt==19 && fi>1) ));
6927         } else if(g==endd){
6928             if(is_neg){
6929                 x= -x;
6930             }
6931             if(!(l <= x && x <= r))
6932             {
6933                 cerr << l << ' ' << r << ' ' << x << '\n';
6934                 assert(1 == 0);
6935             }
6936             return x;
6937         } else {
6938             assert(false);
6939         }
6940     }
6941 }
6942 }

```

```

6943     }
6944     string readString(int l,int r,char endd){
6945         string ret="";
6946         int cnt=0;
6947         while(true){
6948             char g=getchar();
6949             assert(g!=-1);
6950             if(g==endd){
6951                 break;
6952             }
6953             cnt++;
6954             ret+=g;
6955         }
6956         assert(l<=cnt && cnt<=r);
6957         return ret;
6958     }
6959     long long readIntSp(long long l,long long r){
6960         return readInt(l,r,' ');
6961     }
6962     long long readIntLn(long long l,long long r){
6963         return readInt(l,r,'\n');
6964     }
6965     string readStringLn(int l,int r){
6966         return readString(l,r,'\n');
6967     }
6968     string readStringSp(int l,int r){
6969         return readString(l,r,' ');
6970     }
6971     int sumN=0;
6972     long long A[N],B[N],C[N];
6973     void solve()
6974     {
6975         int N=readInt(2,100000,'\n');
6976         sumN+=N;
6977         assert(sumN<=100000);
6978         long long sumA=0,sumB=0;
6979         for(int i=1;i<=N;i++)
6980         {
6981             if(i==N)
6982                 A[i]=readInt(1,1000000000,'\n');
6983             else
6984                 A[i]=readInt(1,1000000000,' ');
6985             sumA+=A[i];
6986         }
6987         for(int i=1;i<=N;i++)
6988         {
6989             if(i==N)
6990                 B[i]=readInt(1,1000000000,'\n');
6991             else
6992                 B[i]=readInt(1,1000000000,' ');
6993             sumB+=B[i];
6994         }
6995         if(sumA!=sumB)
6996         {
6997             cout<<-1<<'\n';
6998             return;
6999         }
7000         for(int i=1;i<=N;i++)
7001             C[i]=max(A[i],B[i]);
7002         long long ans=0;
7003         for(int i=1;i<=N;i++)
7004             ans+=(C[i]-A[i]);
7005         cout<<ans<<'\n';
7006     }
7007     int main()
7008     {
7009         #ifndef ONLINE_JUDGE
7010         freopen("input.txt", "r", stdin);
7011         freopen("output.txt", "w", stdout);

```

```

7012     #endif
7013     ios_base::sync_with_stdio(false);
7014     cin.tie(NULL),cout.tie(NULL);
7015     int T=readInt(1,20000,'\n');
7016     while(T--)
7017         solve();
7018     assert(getchar()==-1);
7019     cerr << "Time : " << 1000 * ((double)clock()) / (double)CLOCKS_PER_SEC << "ms\n";
7020 }
7021
7022 //ROCKET_PACK
7023 #include <cstdio>
7024 #include <queue>
7025 using namespace std;
7026 typedef long long ll;
7027 int T,n,m,k,x,y,c,e;
7028
7029 struct rocket
7030 {
7031     int h,e;
7032     ll c;
7033
7034     rocket(){}
7035
7036     rocket(int h,ll c,int e):h(h),c(c),e(e) {}
7037 };
7038
7039 struct cmp1
7040 {
7041     bool operator () (rocket x,rocket y)
7042     {
7043         return x.c>y.c;
7044     }
7045 };
7046
7047 struct cmp2
7048 {
7049     bool operator () (rocket x,rocket y)
7050     {
7051         return x.h>y.h;
7052     }
7053 };
7054 priority_queue<rocket,vector<rocket>,cmp1> Q1;
7055 priority_queue<rocket,vector<rocket>,cmp2> Q2;
7056
7057 int main()
7058 {
7059     scanf("%d",&T);
7060     while(T--)
7061     {
7062         scanf("%d%d%d",&n,&m,&k);
7063         while(!Q1.empty()) Q1.pop();
7064         while(!Q2.empty()) Q2.pop();
7065         for(int i=1;i<=k;i++)
7066         {
7067             scanf("%d%d%d%d",&x,&y,&c,&e);
7068             if(x+y) Q2.push(rocket(x+y,c,e));
7069             else Q1.push(rocket(x+y,c,e));
7070         }
7071         while(!Q1.empty())
7072         {
7073             rocket x=Q1.top(),y;
7074             if((ll)x.h+x.e>=n+m)
7075             {
7076                 printf("%lld\n",x.c);
7077                 break;
7078             }
7079             Q1.pop();
7080             if(Q2.empty()) continue;

```



```

7081         y=Q2.top();
7082         while( (ll)x.h+x.e>=y.h)
7083         {
7084             Q2.pop();
7085             y.c+=x.c;
7086             Q1.push(y);
7087             if(Q2.empty()) break;
7088             else y=Q2.top();
7089         }
7090     }
7091 }
7092
7093     return 0;
7094 }
7095
7096 //ROCKET_PACK-EDITOR
7097 #include "bits/stdc++.h"
7098 // #pragma GCC optimize("O3,unroll-loops")
7099 // #pragma GCC target("avx2,bmi,bmi2,lzcnt,popcnt")
7100 using namespace std;
7101 using ll = long long int;
7102 mt19937_64 rng(chrono::high_resolution_clock::now().time_since_epoch().count());
7103
7104 int main()
7105 {
7106     ios::sync_with_stdio(false); cin.tie(0);
7107
7108     int t; cin >> t;
7109     while (t--) {
7110         int n, m, k; cin >> n >> m >> k;
7111         vector<array<ll, 3>> events;
7112         for (int i = 0; i < k; ++i) {
7113             ll x, y, c, e; cin >> x >> y >> c >> e;
7114             events.push_back({x+y, c, e});
7115         }
7116         events.push_back({n+m, 0, 0});
7117         sort(begin(events), end(events));
7118
7119         set<array<ll, 2>> active = {{0, 1}}, to_rem = {{1, 0}};
7120         ll ans = 0;
7121         for (auto [s, c, e] : events) {
7122             while (!to_rem.empty()) {
7123                 auto [tm, val] = *to_rem.begin();
7124                 if (tm <= s) {
7125                     to_rem.erase(to_rem.begin());
7126                     active.erase({val, tm});
7127                 }
7128                 else break;
7129             }
7130             assert(!active.empty());
7131             auto it = *active.begin();
7132             if (s == n+m) {
7133                 ans = it[0];
7134                 break;
7135             }
7136             ll nxt = it[0] + c;
7137             // from s+1 to s+e
7138             active.insert({nxt, s+e+1});
7139             to_rem.insert({s+e+1, nxt});
7140         }
7141         cout << ans << '\n';
7142     }
7143 }
7144
7145 //QUEENBL
7146 #include "bits/stdc++.h"
7147 // #pragma GCC optimize("O3,unroll-loops")
7148 // #pragma GCC target("avx2,bmi,bmi2,lzcnt,popcnt")
7149 using namespace std;

```

```

7150 using ll = long long int;
7151 mt19937_64 rng(chrono::high_resolution_clock::now().time_since_epoch().count());
7152
7153 int ans[8][8];
7154
7155 int main()
7156 {
7157     ios::sync_with_stdio(false); cin.tie(0);
7158
7159     auto attack = [] (int x, int y, int a, int b) {
7160         // does a queen at (x, y) attack (a, b) ?
7161         return x == a or y == b or x+y == a+b or x-y == a-b;
7162     };
7163
7164     int t; cin >> t;
7165     while (t--) {
7166         int x, y; cin >> x >> y;
7167         --x, --y;
7168
7169         int mn = 4;
7170         for (int i = 0; i < 64; ++i) {
7171             int x1 = i/8, y1 = i%8;
7172             if (attack(x1, y1, x, y)) continue;
7173             for (int j = 0; j < 64; ++j) {
7174                 int x2 = j/8, y2 = j%8;
7175                 if (attack(x2, y2, x, y)) continue;
7176                 for (int k = 0; k < 64; ++k) {
7177                     int x3 = k/8, y3 = k%8;
7178                     if (attack(x3, y3, x, y)) continue;
7179
7180                     int ct = size(set{i, j, k});
7181                     if (mn <= ct) continue;
7182
7183                     bool good = 1;
7184                     for (int dx : {-1, 0, 1}) for (int dy : {-1, 0, 1}) {
7185                         if (dx == 0 and dy == 0) continue;
7186                         int nx = x+dx, ny = y+dy;
7187                         if (nx < 0 or ny < 0 or nx == 8 or ny == 8) continue;
7188
7189                         good &= attack(x1, y1, nx, ny) || attack(x2, y2, nx, ny) ||
7190                             attack(x3, y3, nx, ny);
7191                     }
7192                     if (!good) continue;
7193                     memset(ans, 0, sizeof ans);
7194                     ans[x][y] = 1;
7195                     ans[x1][y1] = ans[x2][y2] = ans[x3][y3] = 2;
7196                     mn = ct;
7197                 }
7198             }
7199             for (int i = 0; i < 8; ++i) {
7200                 for (int j = 0; j < 8; ++j) cout << ans[i][j] << " \n"[j == 7];
7201             }
7202         }
7203     }
7204
7205     //MAKEIT1
7206     #include <bits/stdc++.h>
7207     using namespace std;
7208
7209     /*
7210     -----Input Checker-----
7211     */
7212
7213     long long readInt(long long l, long long r, char endd) {
7214         long long x=0;
7215         int cnt=0;
7216         int fi=-1;

```

```

7218     bool is_neg=false;
7219     while(true){
7220         char g=getchar();
7221         if(g=='-'){
7222             assert(fi==-1);
7223             is_neg=true;
7224             continue;
7225         }
7226         if('0'<=g && g<='9'){
7227             x*=10;
7228             x+=g-'0';
7229             if(cnt==0){
7230                 fi=g-'0';
7231             }
7232             cnt++;
7233             assert(fi!=0 || cnt==1);
7234             assert(fi!=0 || is_neg==false);
7235
7236             assert(!(cnt>19 || ( cnt==19 && fi>1) ));
7237         } else if(g==endd){
7238             if(is_neg){
7239                 x= -x;
7240             }
7241
7242             if(!(l <= x && x <= r))
7243             {
7244                 cerr << l << ' ' << r << ' ' << x << '\n';
7245                 assert(1 == 0);
7246             }
7247
7248             return x;
7249         } else {
7250             assert(false);
7251         }
7252     }
7253 }
7254 string readString(int l,int r,char endd){
7255     string ret="";
7256     int cnt=0;
7257     while(true){
7258         char g=getchar();
7259         assert(g!=-1);
7260         if(g==endd){
7261             break;
7262         }
7263         cnt++;
7264         ret+=g;
7265     }
7266     assert(l<=cnt && cnt<=r);
7267     return ret;
7268 }
7269 long long readIntSp(long long l,long long r){
7270     return readInt(l,r,' ');
7271 }
7272 long long readIntLn(long long l,long long r){
7273     return readInt(l,r,'\n');
7274 }
7275 string readStringLn(int l,int r){
7276     return readString(l,r,'\n');
7277 }
7278 string readStringSp(int l,int r){
7279     return readString(l,r,' ');
7280 }
7281
7282
7283 /*
7284 -----Main code starts here-----
7285 */
7286

```

```

7287     const int MAX_T = 1e5;
7288     const int MAX_N = 1e5;
7289     const int MAX_SUM_LEN = 1e5;
7290
7291     #define fast_ios_base::sync_with_stdio(0); cin.tie(0); cout.tie(0)
7292     #define ff first
7293     #define ss second
7294     #define mp make_pair
7295     #define ll long long
7296     #define rep(i,n) for(int i=0;i<n;i++)
7297     #define rev(i,n) for(int i=n;i>=0;i--)
7298     #define rep_a(i,a,n) for(int i=a;i<n;i++)
7299     #define pb push_back
7300     #define int ll
7301
7302     int sum_n = 0, sum_m = 0;
7303     int max_n = 0, max_m = 0;
7304     int yess = 0;
7305     int nos = 0;
7306     int total_ops = 0;
7307     ll mod = 998244353;
7308
7309     const ll MX=10000005;
7310     ll lp[MX]={0};
7311     vector<int> pr;
7312
7313     void pre(){
7314         rep_a(i,2,MX){
7315             if(lp[i]==0){
7316                 lp[i]=i;
7317                 pr.pb(i);
7318             }
7319             for(int j=0;j<(int)pr.size() && pr[j]<=lp[i] && i*pr[j]<MX;j++){
7320                 lp[i*pr[j]]=pr[j];
7321             }
7322         }
7323     }
7324
7325     vector<vector<pair<int,int> > >g;
7326     vector<int> dp, col;
7327     bool poss;
7328     void dfs(int curr){
7329         col[curr]=1;
7330         dp[curr] = 1;
7331         for(auto h:g[curr]){
7332             if(col[h.ff]==1){
7333                 poss = 0;
7334                 return;
7335             }
7336             else if(!col[h.ff]){
7337                 dfs(h.ff);
7338             }
7339             dp[curr] += (h.ss*dp[h.ff])%mod;
7340             dp[curr] %= mod;
7341         }
7342         col[curr]=2;
7343
7344         if(curr==1) dp[curr] = 0;
7345     }
7346
7347     void solve(){
7348         int n = readIntSp(1, 1e6);
7349         int m = readIntLn(0, 1e5);
7350
7351         g.assign(MX, vector<pair<int,int> >());
7352         dp.assign(MX,0);
7353         col.assign(MX,0);
7354     }
7355 
```

```

7356     int cnt, tmp;
7357
7358
7359     int a,b;
7360     rep(i,m){
7361         a = readIntSp(1,1e6);
7362         b = readIntLn(1,1e6);
7363
7364         while(b>1){
7365             cnt = 0;
7366             tmp = lp[b];
7367             while(b%tmp==0){
7368                 b/=tmp;
7369                 cnt++;
7370             }
7371
7372             g[a].pb(mp(tmp, cnt));
7373         }
7374     }
7375
7376     poss = 1;
7377     int ans = 0;
7378     // for(auto h:g[13]) cout<<h.ff<<" "<<h.ss<<'\n';
7379     while(n>1){
7380         cnt = 0;
7381         tmp = lp[n];
7382         while(n%tmp==0){
7383             n/=tmp;
7384             cnt++;
7385         }
7386
7387         dfs(tmp);
7388         if(!poss) break;
7389
7390         //cout<<tmp<<" "<<dp[tmp]<<'\n';
7391         ans += (cnt*dp[tmp])%mod;
7392         ans%=mod;
7393     }
7394
7395     // rep(i, 1e6) cout<<dp[i]<<'\n';
7396
7397     if(poss) cout<<ans<<'\n';
7398     else cout<<-1<<'\n';
7399
7400
7401 }
7402
7403 signed main()
7404 {
7405
7406     #ifndef ONLINE_JUDGE
7407     freopen("input.txt", "r" , stdin);
7408     freopen("output.txt", "w" , stdout);
7409     #endif
7410     fast;
7411
7412     pre();
7413     int t = 1;
7414
7415     for(int i=1;i<=t;i++)
7416     {
7417         solve();
7418     }
7419
7420     assert(getchar() == -1);
7421
7422     cerr<<"SUCCESS\n";
7423
7424

```

```

7425
7426     cerr << "Time : " << 1000 * ((double)clock()) / (double)CLOCKS_PER_SEC << "ms\n";
7427 }
7428
7429 //MAKEIT1-EDITOR
7430 #include "bits/stdc++.h"
7431 // #pragma GCC optimize("O3,unroll-loops")
7432 // #pragma GCC target("avx2,bmi,bmi2,lzcnt,popcnt")
7433 using namespace std;
7434 using ll = long long int;
7435 mt19937_64 rng(chrono::high_resolution_clock::now().time_since_epoch().count());
7436
7437 int main()
7438 {
7439     ios::sync_with_stdio(false); cin.tie(0);
7440
7441     const int LIM = 1e6 + 5, mod = 998244353;
7442     vector<int> spf(LIM);
7443     for (int i = LIM-1; i >= 2; --i) {
7444         for (int j = i; j < LIM; j += i) spf[j] = i;
7445     }
7446
7447     int n, m; cin >> n >> m;
7448     vector<vector<int>> adj(LIM);
7449     for (int i = 0; i < m; ++i) {
7450         int a, b; cin >> a >> b;
7451         while (b > 1) {
7452             int p = spf[b];
7453             b /= p;
7454             adj[a].push_back(p);
7455         }
7456     }
7457     int ans = 0, bad = 0;
7458     vector<int> cost(LIM, -1), mark(LIM);
7459     auto calc = [&] (const auto &self, int u) -> ll {
7460         if (cost[u] != -1) return cost[u];
7461         if (mark[u] == 1 or bad) {
7462             bad = 1;
7463             return 0;
7464         }
7465         mark[u] = 1;
7466         ll ret = 1;
7467         for (int v : adj[u]) {
7468             ret += self(self, v);
7469             ret %= mod;
7470         }
7471         mark[u] = 2;
7472         return cost[u] = ret;
7473     };
7474
7475     while (n > 1) {
7476         int p = spf[n];
7477         ans += calc(calc, p); ans %= mod;
7478         n /= p;
7479     }
7480     if (bad) cout << -1 << '\n';
7481     else cout << ans << '\n';
7482 }
7483
7484 //KLIP
7485 #include <bits/stdc++.h>
7486 #include <ext/pb_ds/assoc_container.hpp>
7487 #include <ext/pb_ds/tree_policy.hpp>
7488 using namespace std;
7489 using namespace __gnu_pbds;
7490 #define int long long int
7491 #define ordered_set tree<int, nuint_type, less<int>,
7492 rb_tree_tag, tree_order_statistics_node_update>
7493 mt19937

```

```

rng(std::chrono::duration_cast<std::chrono::nanoseconds>(chrono::high_resolution_clock::now().time_since_epoch()).count());
7493 #define mp make_pair
7494 #define pb push_back
7495 #define F first
7496 #define S second
7497 const int N=1000005;
7498 #define M 1000000007
7499 #define BINF 1e16
7500 #define init(arr,val) memset(arr,val,sizeof(arr))
7501 #define MAXN 1750001
7502 #define deb(xx) cout << #xx << " " << xx << "\n";
7503 const int LG = 22;
7504
7505
7506 void solve() {
7507
7508     int n, k;
7509     cin >> n >> k;
7510     string s;
7511     cin >> s;
7512
7513     vector<int>v(n, 0);
7514     for(int i = 0; i < n; i++) {
7515         v[i] = s[i] - '0';
7516     }
7517     queue<int>q;
7518     for(int i = 0; i < n - k + 1; i++) {
7519         while(q.size() > 0 and q.front() < i) {
7520             q.pop();
7521         }
7522         v[i] = (v[i] + q.size()) % 2;
7523         if(v[i] == 0) continue;
7524         v[i] = (v[i] + 1) % 2;
7525         q.push(i + k - 1);
7526     }
7527
7528
7529     for(int i = n - k + 1; i < n; i++) {
7530         while(q.size() > 0 and q.front() < i) {
7531             q.pop();
7532         }
7533         v[i] = (v[i] + q.size()) % 2;
7534     }
7535     string ans(n, '0');
7536     for(int i = 0; i < n; i++) {
7537         ans[i] = ('0' + v[i]);
7538     }
7539
7540     cout << ans << endl;
7541
7542 }
7543
7544
7545 #undef int
7546 int main() {
7547     #define int long long int
7548     ios_base::sync_with_stdio(false);
7549     cin.tie(0);
7550     cout.tie(0);
7551     #ifndef ONLINE_JUDGE
7552         freopen("input.txt", "r", stdin);
7553         freopen("optput.txt", "w", stdout);
7554     #endif
7555
7556
7557     int T;
7558     cin >> T;
7559

```

```

7560     for(int tc = 1; tc <= T; tc++){
7561         // cout << "Case #" << tc << ": ";
7562         solve();
7563     }
7564
7565
7566     return 0;
7567
7568 }
7569
7570 //FLIPFAIL
7571 #include <bits/stdc++.h>
7572 using namespace std;
7573
7574 int main() {
7575     //freopen("inpl.in", "r", stdin);
7576     //freopen("inpl.out", "w", stdout);
7577     int t;
7578     cin >> t;
7579     while(t--) {
7580         string s;
7581         cin >> s;
7582         int cnt = 0, ncnt = 0;
7583         for(int i = 0; i < s.size(); i++) cnt += (s[i] == '0');
7584         if(!cnt) {
7585             cout << "0\n";
7586             continue;
7587         }
7588         for(int i = 0; i < cnt; i++) ncnt += (s[i] == '0');
7589         int ans = cnt - ncnt;
7590         for(int i = cnt; i < s.size(); i++) {
7591             ncnt -= (s[i - cnt] == '0');
7592             ncnt += (s[i] == '0');
7593             ans = min(ans, cnt - ncnt);
7594         }
7595         cout << ans << "\n";
7596     }
7597 }
7598
7599 //FLIPFAIL-EDITOR
7600 #include "bits/stdc++.h"
7601 // #pragma GCC optimize("O3,unroll-loops")
7602 // #pragma GCC target("avx2,bmi,bmi2,lzcnt,popcnt")
7603 using namespace std;
7604 using ll = long long int;
7605 mt19937_64 rng(chrono::high_resolution_clock::now().time_since_epoch().count());
7606
7607 int main()
7608 {
7609     ios::sync_with_stdio(false); cin.tie(0);
7610
7611     int t; cin >> t;
7612     while (t--) {
7613         string s; cin >> s;
7614         int n = s.size();
7615         int ans = n;
7616         vector<int> pref(n+2);
7617         for (int i = 1; i <= n; ++i) {
7618             pref[i] = s[i-1] == '1';
7619             pref[i] += pref[i-1];
7620         }
7621         int z = count(begin(s), end(s), '0');
7622         auto get = [&] (int i, int j) {if (j < i) return 0; return pref[j] - pref[i-1];};
7623
7624         for (int i = 0; i <= n; ++i) {
7625             // flip upto i
7626             if (i <= z) ans = min(ans, pref[z]);
7627             else ans = min(ans, get(i-z+1, i));
7628         }

```



```

7629         cout << ans << '\n';
7630
7631     }
7632 }
7633
7634 //NILXOR
7635 #include "bits/stdc++.h"
7636 // #pragma GCC optimize("O3,unroll-loops")
7637 // #pragma GCC target("avx2,bmi,bmi2,lzcnt,popcnt")
7638 using namespace std;
7639 using ll = long long int;
7640 mt19937_64 rng(chrono::high_resolution_clock::now().time_since_epoch().count());
7641
7642 int main()
7643 {
7644     ios::sync_with_stdio(false); cin.tie(0);
7645
7646     int t; cin >> t;
7647     while (t--) {
7648         int n, x; cin >> n >> x;
7649         int ans = 0, ct = 31 - __builtin_popcount(x);
7650
7651         for (int bit = 30; bit >= 0; --bit) {
7652             if ((x >> bit) & 1) continue;
7653             --ct;
7654             if ((~n >> bit) & 1) continue;
7655
7656             // Let this be the first bit that is strictly less than n
7657             ans += 1 << ct;
7658         }
7659         cout << ans << '\n';
7660     }
7661 }
7662
7663 //NCOPIES
7664 #ifdef WTSH
7665     #include <wtsh.h>
7666 #else
7667     #include <bits/stdc++.h>
7668     using namespace std;
7669     #define dbg(...)
7670 #endif
7671
7672 #define int long long
7673 #define endl "\n"
7674 #define sz(w) (int)(w.size())
7675 using pii = pair<int, int>;
7676
7677 const int mod = 998244353;
7678
7679 // ----- Input Checker Start -----
7680
7681 long long readInt(long long l, long long r, char endd)
7682 {
7683     long long x = 0;
7684     int cnt = 0, fi = -1;
7685     bool is_neg = false;
7686     while(true)
7687     {
7688         char g = getchar();
7689         if(g == '-')
7690         {
7691             assert(fi == -1);
7692             is_neg = true;
7693             continue;
7694         }
7695         if('0' <= g && g <= '9')
7696         {
7697             x *= 10;

```

```

7698         x += g - '0';
7699         if(cnt == 0)
7700             fi = g - '0';
7701         cnt++;
7702         assert(fi != 0 || cnt == 1);
7703         assert(fi != 0 || is_neg == false);
7704         assert(!(cnt > 19 || (cnt == 19 && fi > 1)));
7705     }
7706     else if(g == endd)
7707     {
7708         if(is_neg)
7709             x = -x;
7710         if(!(l <= x && x <= r))
7711         {
7712             cerr << "L: " << l << ", R: " << r << ", Value Found: " << x << '\n';
7713             assert(false);
7714         }
7715         return x;
7716     }
7717     else
7718     {
7719         assert(false);
7720     }
7721 }
7722 }
7723
7724 string readString(int l, int r, char endd)
7725 {
7726     string ret = "";
7727     int cnt = 0;
7728     while(true)
7729     {
7730         char g = getchar();
7731         assert(g != -1);
7732         if(g == endd)
7733             break;
7734         cnt++;
7735         ret += g;
7736     }
7737     assert(l <= cnt && cnt <= r);
7738     return ret;
7739 }
7740
7741 long long readIntSp(long long l, long long r) { return readInt(l, r, ' '); }
7742 long long readIntLn(long long l, long long r) { return readInt(l, r, '\n'); }
7743 string readStringSp(int l, int r) { return readString(l, r, ' '); }
7744 string readStringLn(int l, int r) { return readString(l, r, '\n'); }
7745 void readEOF() { assert(getchar() == EOF); }
7746
7747 vector<int> readVectorInt(int n, long long l, long long r)
7748 {
7749     vector<int> a(n);
7750     for(int i = 0; i < n - 1; i++)
7751         a[i] = readIntSp(l, r);
7752     a[n - 1] = readIntLn(l, r);
7753     return a;
7754 }
7755
7756 // ----- Input Checker End -----
7757
7758 int sumN = 0;
7759
7760 void solve()
7761 {
7762     int n = readIntSp(1, 1e5);
7763     sumN += n;
7764     int m = readIntLn(1, 1e5);
7765     string a = readStringLn(n, n);
7766     assert(*min_element(a.begin(), a.end()) >= '0' and *max_element(a.begin(), a.end())

```

```

7767     <= '1');
7768     int S = count(a.begin(), a.end(), '1');
7769     if(S == 0)
7770         cout << n * m << endl;
7771     else if(S * m % 2 == 1)
7772         cout << 0 << endl;
7773     else
7774     {
7775         string b = a;
7776         if(m % 2 == 0)
7777             b += a, S += S;
7778         int cur = 0, ans = 0;
7779         for(int i = 0; i < sz(b); i++)
7780         {
7781             cur += b[i] - '0';
7782             if(2 * cur == S)
7783                 ans++;
7784             cout << ans << endl;
7785         }
7786     }
7787
7788 int32_t main()
7789 {
7790     ios::sync_with_stdio(0);
7791     cin.tie(0);
7792     int T = readIntLn(1, 1e5);
7793     for(int tc = 1; tc <= T; tc++)
7794     {
7795         // cout << "Case #" << tc << ": ";
7796         solve();
7797     }
7798     readEOF();
7799     assert(sumN <= 2e5);
7800     return 0;
7801 }
7802
7803 //DIFFMED
7804 //Utkarsh.25dec
7805 #include <bits/stdc++.h>
7806 #define ll long long int
7807 #define pb push_back
7808 #define mp make_pair
7809 #define mod 1000000007
7810 #define vl vector<ll>
7811 #define all(c) (c).begin(), (c).end()
7812 using namespace std;
7813 ll power(ll a, ll b) {ll res=1; a%=mod; assert(b>=0);
7814 for(; b>=1; b--){if(b&1) res=res*a%mod; a=a*a%mod;} return res;}
7815 ll modInverse(ll a){return power(a, mod-2);}
7816 const int N=500023;
7817 bool vis[N];
7818 vector<int> adj[N];
7819 long long readInt(long long l, long long r, char endd){
7820     long long x=0;
7821     int cnt=0;
7822     int fi=-1;
7823     bool is_neg=false;
7824     while(true){
7825         char g=getchar();
7826         if(g=='-'){
7827             assert(fi==-1);
7828             is_neg=true;
7829             continue;
7830         }
7831         if('0'<=g && g<='9'){
7832             x*=10;
7833             x+=g-'0';
7834             if(cnt==0){

```

```

7834         fi=g-'0';
7835     }
7836     cnt++;
7837     assert(fi!=0 || cnt==1);
7838     assert(fi!=0 || is_neg==false);
7839
7840     assert(!(cnt>19 || ( cnt==19 && fi>1) ));
7841 } else if(g==endd){
7842     if(is_neg){
7843         x= -x;
7844     }
7845
7846     if(!(l <= x && x <= r))
7847     {
7848         cerr << l << ' ' << r << ' ' << x << '\n';
7849         assert(1 == 0);
7850     }
7851
7852     return x;
7853 } else {
7854     assert(false);
7855 }
7856 }
7857 }
7858 string readString(int l,int r,char endd){
7859     string ret="";
7860     int cnt=0;
7861     while(true){
7862         char g=getchar();
7863         assert(g!=-1);
7864         if(g==endd){
7865             break;
7866         }
7867         cnt++;
7868         ret+=g;
7869     }
7870     assert(l<=cnt && cnt<=r);
7871     return ret;
7872 }
7873 long long readIntSp(long long l,long long r){
7874     return readInt(l,r,' ');
7875 }
7876 long long readIntLn(long long l,long long r){
7877     return readInt(l,r,'\n');
7878 }
7879 string readStringLn(int l,int r){
7880     return readString(l,r,'\n');
7881 }
7882 string readStringSp(int l,int r){
7883     return readString(l,r,' ');
7884 }
7885 int sumN=0;
7886 void solve()
7887 {
7888     int N=readInt(2,1000,'\n');
7889     sumN+=N;
7890     assert(sumN<=1000);
7891     int low=1,high=N;
7892     for(int i=1;i<=N;i++)
7893     {
7894         if(i%2==1)
7895             cout<<(high--)<<' ';
7896         else
7897             cout<<(low++)<<' ';
7898     }
7899     cout<<'\n';
7900 }
7901 int main()
7902 {

```

```

7903     #ifndef ONLINE_JUDGE
7904     freopen("input.txt", "r", stdin);
7905     freopen("output.txt", "w", stdout);
7906     #endif
7907     ios_base::sync_with_stdio(false);
7908     cin.tie(NULL), cout.tie(NULL);
7909     int T=readInt(1,30,'\n');
7910     while(T--)
7911         solve();
7912     assert(getchar() == -1);
7913     cerr << "Time : " << 1000 * ((double)clock()) / (double)CLOCKS_PER_SEC << "ms\n";
7914 }
7915
7916 //DIFFER
7917 #ifndef _GLIBCXX_NO_ASSERT
7918 #include <cassert>
7919 #endif
7920 #include <cctype>
7921 #include <cerrno>
7922 #include <cfloat>
7923 #include <ciso646>
7924 #include <climits>
7925 #include <locale>
7926 #include <cmath>
7927 #include <csetjmp>
7928 #include <csignal>
7929 #include <cstdarg>
7930 #include <cstddef>
7931 #include <cstdio>
7932 #include <cstdlib>
7933 #include <cstring>
7934 #include <ctime>
7935 #include <algorithm>
7936 #include <bitset>
7937 #include <complex>
7938 #include <deque>
7939 #include <exception>
7940 #include <fstream>
7941 #include <functional>
7942 #include <iomanip>
7943 #include <ios>
7944 #include <iosfwd>
7945 #include <iostream>
7946 #include <istream>
7947 #include <iterator>
7948 #include <limits>
7949 #include <list>
7950 #include <locale>
7951 #include <map>
7952 #include <memory>
7953 #include <new>
7954 #include <numeric>
7955 #include <ostream>
7956 #include <queue>
7957 #include <set>
7958 #include <sstream>
7959 #include <stack>
7960 #include <stdexcept>
7961 #include <streambuf>
7962 #include <string>
7963 #include <typeinfo>
7964 #include <utility>
7965 #include <valarray>
7966 #include <vector>
7967
7968 #if __cplusplus >= 201103L
7969 #include <array>
7970 #include <atomic>
7971 #include <chrono>

```

```

7972 #include <condition_variable>
7973 #include <forward_list>
7974 #include <future>
7975 #include <initializer_list>
7976 #include <mutex>
7977 #include <random>
7978 #include <ratio>
7979 #include <regex>
7980 #include <scoped_allocator>
7981 #include <system_error>
7982 #include <thread>
7983 #include <tuple>
7984 #include <typeindex>
7985 #include <type_traits>
7986 #include <unordered_map>
7987 #include <unordered_set>
7988 #endif
7989
7990 #define FR(i, b) for (int i = 0; i < int(b); i++)
7991 #define FOR(i, a, b) for (int i = int(a); i < int(b); i++)
7992 #define all(x) begin(x), end(x)
7993 #define sz(x) int((x).size())
7994
7995 using ll = long long;
7996
7997 using namespace std;
7998
7999
8000 void solve() {
8001     int n, m;
8002     cin >> n >> m;
8003
8004     vector<vector<ll>> a(n, vector<ll>(m, 0));
8005     vector<ll> tot(n, 0);
8006     for (int i = 0; i < n; i++) {
8007         for (int j = 0; j < m; j++) {
8008             cin >> a[i][j];
8009             tot[i] += a[i][j];
8010         }
8011         for (int j = 0; j < m; j++) {
8012             a[i][j] = m*a[i][j] - tot[i];
8013         }
8014     }
8015     ll res = 0;
8016     for (int j = 0; j < m; j++) {
8017         ll mn = 0, mx = 0;
8018         for (int i = 0; i < n; i++) {
8019             if (a[i][j] < 0) {
8020                 mn -= a[i][j];
8021             }
8022             if (a[i][j] > 0) {
8023                 mx += a[i][j];
8024             }
8025         }
8026         res = max(res, max(mn, mx));
8027     }
8028     cout << res << '\n';
8029 }
8030 int main() {
8031     ios_base::sync_with_stdio(false);
8032     cin.tie(0);
8033     int T;
8034     cin >> T;
8035     FOR(t, 1, T+1) {
8036         solve();
8037     }
8038     return 0;
8039 }
8040

```

```

8041 //ADJXOR2
8042 #ifndef WTSH
8043     #include <wtsh.h>
8044 #else
8045     #include <bits/stdc++.h>
8046     using namespace std;
8047     #define dbg(...)
8048 #endif
8049
8050 #define int long long
8051 #define endl "\n"
8052 #define sz(w) (int)(w.size())
8053 using pii = pair<int, int>;
8054
8055 const int mod = 998244353;
8056
8057 // ----- Input Checker Start -----
8058
8059 long long readInt(long long l, long long r, char endd)
8060 {
8061     long long x = 0;
8062     int cnt = 0, fi = -1;
8063     bool is_neg = false;
8064     while(true)
8065     {
8066         char g = getchar();
8067         if(g == '-')
8068         {
8069             assert(fi == -1);
8070             is_neg = true;
8071             continue;
8072         }
8073         if('0' <= g && g <= '9')
8074         {
8075             x *= 10;
8076             x += g - '0';
8077             if(cnt == 0)
8078                 fi = g - '0';
8079             cnt++;
8080             assert(fi != 0 || cnt == 1);
8081             assert(fi != 0 || is_neg == false);
8082             assert(!(cnt > 19 || (cnt == 19 && fi > 1)));
8083         }
8084         else if(g == endd)
8085         {
8086             if(is_neg)
8087                 x = -x;
8088             if(!(l <= x && x <= r))
8089             {
8090                 cerr << "L: " << l << ", R: " << r << ", Value Found: " << x << '\n';
8091                 assert(false);
8092             }
8093             return x;
8094         }
8095         else
8096         {
8097             assert(false);
8098         }
8099     }
8100 }
8101
8102 string readString(int l, int r, char endd)
8103 {
8104     string ret = "";
8105     int cnt = 0;
8106     while(true)
8107     {
8108         char g = getchar();
8109         assert(g != -1);

```

```

8110         if(g == endd)
8111             break;
8112         cnt++;
8113         ret += g;
8114     }
8115     assert(l <= cnt && cnt <= r);
8116     return ret;
8117 }
8118
8119 long long readIntSp(long long l, long long r) { return readInt(l, r, ' '); }
8120 long long readIntLn(long long l, long long r) { return readInt(l, r, '\n'); }
8121 string readStringSp(int l, int r) { return readString(l, r, ' '); }
8122 string readStringLn(int l, int r) { return readString(l, r, '\n'); }
8123 void readEOF() { assert(getchar() == EOF); }
8124
8125 vector<int> readVectorInt(int n, long long l, long long r)
8126 {
8127     vector<int> a(n);
8128     for(int i = 0; i < n - 1; i++)
8129         a[i] = readIntSp(l, r);
8130     a[n - 1] = readIntLn(l, r);
8131     return a;
8132 }
8133
8134 // ----- Input Checker End -----
8135
8136 int sumN = 0;
8137
8138 void solve()
8139 {
8140     int n = readIntSp(1, 1e5);
8141     int x = readIntLn(1, 1e9);
8142     vector<int> a = readVectorInt(n, 1, 1e9);
8143     sumN += n;
8144     vector<array<int, 2>> dp(n);
8145     for(int i = 1; i < n; i++)
8146     {
8147         dp[i][0] = max(dp[i - 1][0] + (a[i - 1] ^ a[i]), dp[i - 1][1] + ((a[i - 1] + x)
            ^ a[i]));
8148         dp[i][1] = max(dp[i - 1][0] + (a[i - 1] ^ (a[i] + x)), dp[i - 1][1] + ((a[i -
            1] + x) ^ (a[i] + x)));
8149     }
8150     cout << max(dp[n - 1][0], dp[n - 1][1]) << endl;
8151 }
8152
8153 int32_t main()
8154 {
8155     ios::sync_with_stdio(0);
8156     cin.tie(0);
8157     int T = readIntLn(1, 1e5);
8158     for(int tc = 1; tc <= T; tc++)
8159     {
8160         // cout << "Case #" << tc << ": ";
8161         solve();
8162     }
8163     readEOF();
8164     assert(sumN <= 2e5);
8165     return 0;
8166 }
8167
8168 //CCGAME
8169 #include <bits/stdc++.h>
8170 using namespace std;
8171
8172 #define int long long
8173 #define ii pair<int,int>
8174 #define fi first
8175 #define se second
8176

```



```

8177 #define puf push_front
8178 #define pof pop_front
8179 #define pub push_back
8180 #define pob pop_back
8181 #define lb lower_bound
8182 #define ub upper_bound
8183
8184 #define rep(x,s,e) for (int x=(s)-((s)>(e));x!=(e)-((s)>(e));((s)<(e))?x++:x--)
8185 #define all(x) (x).begin(),(x).end()
8186 #define sz(x) (int) (x).size()
8187
8188 mt19937 rng(chrono::system_clock::now().time_since_epoch().count());
8189
8190 int n;
8191 int arr[105];
8192
8193 signed main(){
8194     cin.tie(0);
8195     cout.tie(0);
8196     cin.sync_with_stdio(false);
8197
8198     int TC;
8199     cin>>TC;
8200     while (TC--){
8201         cin>>n;
8202         rep(x,1,n+1) cin>>arr[x];
8203
8204         int res=0;
8205         rep(x,1,n+1) if (arr[x]&1) res^=(n-x);
8206         if (res==0) cout<<"Cook"<<endl;
8207         else cout<<"Chef"<<endl;
8208     }
8209 }
8210
8211 //PREFSUFANDOR
8212 #include <bits/stdc++.h>
8213 using namespace std;
8214
8215 #define int long long
8216 #define endl "\n"
8217 #define sz(w) (int) (w.size())
8218
8219 const int mod = 998244353;
8220
8221 void solve()
8222 {
8223     int n, k; cin >> n >> k;
8224     vector<int> a(n);
8225     for(int &x: a) cin >> x;
8226     int ans = 1;
8227     for(int i = 0; i < k; i++)
8228     {
8229         vector<int> bit(n);
8230         for(int j = 0; j < n; j++)
8231             bit[j] = (a[j] >> i) & 1;
8232         int cnt0 = count(bit.begin(), bit.end(), 0);
8233         int res = cnt0 + 1;
8234         for(int j = 0; j < n; j++)
8235         {
8236             if(bit[j] == 0)
8237                 cnt0--;
8238             else
8239                 res += cnt0 + 1;
8240         }
8241         ans = ans * (res % mod) % mod;
8242     }
8243     cout << ans << endl;
8244 }
8245

```

```

8246 int32_t main()
8247 {
8248     ios::sync_with_stdio(0);
8249     cin.tie(0);
8250     int T; cin >> T;
8251     while(T--)
8252         solve();
8253     return 0;
8254 }
8255
8256 //START50-EXTERNALSITE
8257 //HIGHFREQ
8258 #include <bits/stdc++.h>
8259 #define ll long long int
8260 #define pb push_back
8261 #define mp make_pair
8262 #define mod 1000000007
8263 #define vl vector <ll>
8264 #define all(c) (c).begin(), (c).end()
8265 using namespace std;
8266 ll power(ll a, ll b) {ll res=1; a%=mod; assert(b>=0);
8267     for(; b; b>>=1) {if(b&1) res=res*a%mod; a=a*a%mod;} return res;}
8268 ll modInverse(ll a) {return power(a, mod-2);}
8269 const int N=500023;
8270 bool vis[N];
8271 vector <int> adj[N];
8272 long long readInt(long long l, long long r, char endd) {
8273     long long x=0;
8274     int cnt=0;
8275     int fi=-1;
8276     bool is_neg=false;
8277     while(true) {
8278         char g=getchar();
8279         if(g=='-') {
8280             assert(fi==--1);
8281             is_neg=true;
8282             continue;
8283         }
8284         if('0'<=g && g<='9') {
8285             x*=10;
8286             x+=g-'0';
8287             if(cnt==0) {
8288                 fi=g-'0';
8289             }
8290             cnt++;
8291             assert(fi!=0 || cnt==1);
8292             assert(fi!=0 || is_neg==false);
8293
8294             assert(!(cnt>19 || ( cnt==19 && fi>1) ));
8295         } else if(g==endd) {
8296             if(is_neg) {
8297                 x= -x;
8298             }
8299
8300             if(!(1 <= x && x <= r))
8301             {
8302                 cerr << l << ' ' << r << ' ' << x << '\n';
8303                 assert(1 == 0);
8304             }
8305             return x;
8306         } else {
8307             assert(false);
8308         }
8309     }
8310 }
8311 string readString(int l, int r, char endd) {
8312     string ret="";
8313     int cnt=0;

```

```

8314     while(true){
8315         char g=getchar();
8316         assert(g!=-1);
8317         if(g==endd){
8318             break;
8319         }
8320         cnt++;
8321         ret+=g;
8322     }
8323     assert(l<=cnt && cnt<=r);
8324     return ret;
8325 }
8326 long long readIntSp(long long l,long long r){
8327     return readInt(l,r,' ');
8328 }
8329 long long readIntLn(long long l,long long r){
8330     return readInt(l,r,'\n');
8331 }
8332 string readStringLn(int l,int r){
8333     return readString(l,r,'\n');
8334 }
8335 string readStringSp(int l,int r){
8336     return readString(l,r,' ');
8337 }
8338 int sumN=0;
8339 void solve()
8340 {
8341     int n=readInt(2,100000,'\n');
8342     sumN+=n;
8343     assert(sumN<=200000);
8344     int A[n+1]={0};
8345     int freq[n+1]={0};
8346     for(int i=1;i<=n;i++)
8347     {
8348         if(i!=n)
8349             A[i]=readInt(1,n,' ');
8350         else
8351             A[i]=readInt(1,n,'\n');
8352         freq[A[i]]++;
8353     }
8354     vector<int> v;
8355     for(int i=1;i<=n;i++)
8356         v.pb(freq[i]);
8357     sort(all(v));
8358     reverse(all(v));
8359     int x=v[0];
8360     int y=v[1];
8361     cout<<max((x+1)/2,y)<<'\n';
8362 }
8363 int main()
8364 {
8365     #ifndef ONLINE_JUDGE
8366     freopen("input.txt", "r", stdin);
8367     freopen("output.txt", "w", stdout);
8368     #endif
8369     ios_base::sync_with_stdio(false);
8370     cin.tie(NULL),cout.tie(NULL);
8371     int T=readInt(1,5000,'\n');
8372     while(T--)
8373         solve();
8374     assert(getchar()==-1);
8375     cerr << "Time : " << 1000 * ((double)clock()) / (double)CLOCKS_PER_SEC << "ms\n";
8376 }
8377
8378 //FARHOSTEL
8379 #include "bits/stdc++.h"
8380 using namespace std;
8381
8382 typedef long long      ll;

```

```

8383 const unsigned int M = 1000000007;
8384
8385 vector<vector<ll>> dp;
8386
8387 ll calcStudents(vector<vector<ll>>& A, vector<vector<ll>>& B){
8388     int n=A.size();
8389     int m=A[0].size();
8390     for(int i=n-2;i>=0;i--){
8391         for(int j=0;j<m;j++){
8392             A[i][j]+=A[i+1][j];
8393         }
8394     }
8395     for(int i=1;i<n;i++){
8396         for(int j=0;j<m;j++){
8397             B[i][j]+=B[i-1][j];
8398         }
8399     }
8400     vector<vector<ll>> dp(n+1, vector<ll>(m+1));
8401     for(int col=m-1;col>=0;col--){
8402         for(int row=n;row>=0;row--){
8403             dp[row][col]=0;
8404             if(col<m-1)dp[row][col]+=dp[row][col+1];
8405             if(row)dp[row][col]+=B[row-1][col];
8406             if(row<n){
8407                 dp[row][col]+=A[row][col];
8408                 dp[row][col]=max(dp[row][col],dp[row+1][col]);
8409             }
8410         }
8411     }
8412     return dp[0][0];
8413 }
8414
8415 int main(){
8416     int t;
8417     cin>>t;
8418     while(t--){
8419         int n, m;
8420         cin>>n>>m;
8421         vector<vector<ll>> A(n,vector<ll>(m));
8422         for(int i=0;i<n;i++){
8423             for(int j=0;j<m;j++){
8424                 cin>>A[i][j];
8425             }
8426         }
8427         vector<vector<ll>> B(n,vector<ll>(m));
8428         for(int i=0;i<n;i++){
8429             for(int j=0;j<m;j++){
8430                 cin>>B[i][j];
8431             }
8432         }
8433         cout<<calcStudents(A,B)<<"\n";
8434     }
8435     return 0;
8436 }
8437
8438 //MONKS
8439 #include <wtsh.h>
8440 #else
8441 #include <bits/stdc++.h>
8442 using namespace std;
8443 #define dbg(...)
8444 #endif
8445
8446 #define int long long
8447 #define endl "\n"
8448 #define sz(w) (int)(w.size())
8449 using pii = pair<int, int>;
8450
8451 const long long INF = 1e18;
8452
8453 const int N = 1e6 + 5;
8454
8455 // ----- Input Checker Start -----

```

```

8452
8453 long long readInt(long long l, long long r, char endd)
8454 {
8455     long long x = 0;
8456     int cnt = 0, fi = -1;
8457     bool is_neg = false;
8458     while(true)
8459     {
8460         char g = getchar();
8461         if(g == '-')
8462         {
8463             assert(fi == -1);
8464             is_neg = true;
8465             continue;
8466         }
8467         if('0' <= g && g <= '9')
8468         {
8469             x *= 10;
8470             x += g - '0';
8471             if(cnt == 0)
8472                 fi = g - '0';
8473             cnt++;
8474             assert(fi != 0 || cnt == 1);
8475             assert(fi != 0 || is_neg == false);
8476             assert(!(cnt > 19 || (cnt == 19 && fi > 1)));
8477         }
8478         else if(g == endd)
8479         {
8480             if(is_neg)
8481                 x = -x;
8482             if(!(l <= x && x <= r))
8483             {
8484                 cerr << "L: " << l << ", R: " << r << ", Value Found: " << x << '\n';
8485                 assert(false);
8486             }
8487             return x;
8488         }
8489         else
8490         {
8491             assert(false);
8492         }
8493     }
8494 }
8495
8496 string readString(int l, int r, char endd)
8497 {
8498     string ret = "";
8499     int cnt = 0;
8500     while(true)
8501     {
8502         char g = getchar();
8503         assert(g != -1);
8504         if(g == endd)
8505             break;
8506         cnt++;
8507         ret += g;
8508     }
8509     assert(l <= cnt && cnt <= r);
8510     return ret;
8511 }
8512
8513 long long readIntSp(long long l, long long r) { return readInt(l, r, ' '); }
8514 long long readIntLn(long long l, long long r) { return readInt(l, r, '\n'); }
8515 string readStringSp(int l, int r) { return readString(l, r, ' '); }
8516 string readStringLn(int l, int r) { return readString(l, r, '\n'); }
8517 void readEOF() { assert(getchar() == EOF); }
8518
8519 vector<int> readVectorInt(int n, long long l, long long r)
8520 {

```

```

8521     vector<int> a(n);
8522     for(int i = 0; i < n - 1; i++)
8523         a[i] = readIntSp(1, r);
8524     a[n - 1] = readIntLn(1, r);
8525     return a;
8526 }
8527
8528 // ----- Input Checker End -----
8529
8530 int sumN = 0;
8531
8532 void solve()
8533 {
8534     int n = readIntLn(1, 1e5);
8535     sumN += n;
8536     vector<int> a = readVectorInt(n, 1, 1e9);
8537     sort(a.begin(), a.end());
8538     vector<int> pref = a;
8539     for(int i = 1; i < n; i++)
8540         pref[i] += pref[i - 1];
8541     for(int i = n - 1; i >= 0; i--)
8542     {
8543         int S = pref.back() - pref[i];
8544         int P = pref[i];
8545         if((i + 1) * a[i] - P <= S)
8546         {
8547             cout << n - (i + 1) << endl;
8548             return;
8549         }
8550     }
8551     assert(false);
8552 }
8553
8554 int32_t main()
8555 {
8556     ios::sync_with_stdio(0);
8557     cin.tie(0);
8558     int T = readIntLn(1, 1e5);
8559     for(int tc = 1; tc <= T; tc++)
8560     {
8561         // cout << "Case #" << tc << ": ";
8562         solve();
8563     }
8564     assert(sumN <= 2e5);
8565     readEOF();
8566     return 0;
8567 }
8568
8569 //MONKS-EDITOR
8570 using namespace std;
8571 #define ll long long
8572
8573 int main() {
8574     ll T;
8575     cin >> T;
8576     while(T--){
8577         ll n;
8578         cin >> n;
8579         vector<ll>a(n);
8580         ll sum=0;
8581         for(ll i=0;i<n;i++){
8582             cin >> a[i];
8583             sum+=a[i];
8584         }
8585         sort(a.begin(),a.end());
8586         if(a[n-1]==a[0]){
8587             cout << 0 << endl;
8588             continue;
8589         }

```

```

8590         ll curr=0;
8591         ll ans=n;
8592         for(ll i=0;i<n;i++){
8593             curr+=a[i];
8594             ll x=sum-curr;
8595             if(x>=a[i]*(i+1)-curr) ans=min(ans,n-i-1);
8596         }
8597         cout << ans << endl;
8598     }
8599     return 0;
8600 }
8601
8602 //ON_OFF
8603 int t;
8604     cin>>t;
8605     while(t--){
8606     {
8607         int n,c=0;
8608         string s,r;
8609         cin>>n;
8610         cin>>s>>r;
8611
8612
8613         for(int j=0;j<n;j++){
8614             {
8615                 if(s[j]!=r[j])
8616                     c++;
8617             }
8618
8619
8620             if(c%2==0)
8621                 cout<<"1"<<"\n";
8622             else
8623                 cout<<"0"<<"\n";
8624
8625 //BFSDFS
8626 #include <bits/stdc++.h>
8627 using namespace std;
8628
8629 const int MAX_T = 1e5;
8630 const int MAX_N = 1e5;
8631 const int MAX_SUM_LEN = 1e5;
8632
8633 #define fast ios_base::sync_with_stdio(0); cin.tie(0); cout.tie(0)
8634 #define ff first
8635 #define ss second
8636 #define mp make_pair
8637 #define ll long long
8638 #define rep(i,n) for(int i=0;i<n;i++)
8639 #define rev(i,n) for(int i=n;i>=0;i--)
8640 #define rep_a(i,a,n) for(int i=a;i<n;i++)
8641 #define pb push_back
8642 #define all(ds) ds.begin(), ds.end()
8643 #define int ll
8644
8645
8646 ll mod = 998244353;
8647
8648 ll po(ll x, ll n ){
8649     ll ans=1;
8650     while(n>0){
8651         if(n&1) ans=(ans*x)%mod;
8652         x=(x*x)%mod;
8653         n/=2;
8654     }
8655     return ans;
8656 }
8657
8658 vector<ll> fac_inv(10);

```

```

8659
8660 void pre() {
8661     fac_inv[0]=1;
8662     rep_a(i,1,10) fac_inv[i]=(i*fac_inv[i-1]);
8663     rep(i,10){
8664         fac_inv[i] = po(fac_inv[i], mod-2);
8665     }
8666 }
8667
8668 vector<vector<int> > adj, lvl_cnt;
8669 vector<int> subsz;
8670
8671 vector<vector<ll> > dfsp, bfsp, comp;
8672 int cnt;
8673 int n;
8674
8675 void chk(int c, int p){
8676     cnt++;
8677     subsz[c]=1;
8678     lvl_cnt[c][0]=1;
8679     for(auto h:adj[c]){
8680         if(h!=p){
8681             chk(h,c);
8682             subsz[c]+=subsz[h];
8683             rep_a(i,1,n) lvl_cnt[c][i]+=lvl_cnt[h][i-1];
8684         }
8685     }
8686 }
8687
8688 //compute DFS-array//
8689
8690 void dfsC(int c, int p){
8691     vector<int> tmp;
8692     for(auto h:adj[c]){
8693         if(h!=p) tmp.pb(h);
8694     }
8695
8696     sort(all(tmp));
8697     int z = tmp.size();
8698     if(z==0) return;
8699
8700     do{
8701         int curr = 1;
8702         for(auto h:tmp){
8703             for(int i=0; i+curr<n; i++){
8704                 dfsp[h][i+curr]+=dfsp[c][i];
8705             }
8706
8707             curr+=subsz[h];
8708         }
8709     }while(next_permutation(all(tmp)));
8710
8711     for(auto h:adj[c]){
8712         if(h!=p){
8713             rep(i,n){
8714                 dfsp[h][i] = ((dfsp[h][i]%mod)*fac_inv[z])%mod;
8715             }
8716
8717             dfsC(h,c);
8718         }
8719     }
8720 }
8721
8722
8723
8724 }
8725
8726
8727 //

```



```

8728
8729 //compute BFS-array////////////////////////////////
8730
8731 void fun(vector<int>&y, vector<int>&y1, int c, int p, int lvl){ //update subtree
position
8732
8733     rep_a(i,y[lvl], n){
8734         comp[c][i]+=bfsp[c][i-y[lvl]+(lvl>0?y1[lvl-1]:0)];
8735     }
8736     for(auto h:adj[c]){
8737         if(h!=p) fun(y,y1,h,c,lvl+1);
8738     }
8739 }
8740
8741 void fun1(int c, int p, ll den){ // copy from temporary array and turn into probability
8742     bfsp[c] = comp[c];
8743     comp[c].assign(n,0);
8744     rep(i,n){
8745         bfsp[c][i] = ((bfsp[c][i]%mod)*den)%mod;
8746     }
8747     for(auto h:adj[c]){
8748         if(h!=p) fun1(h,c,den);
8749     }
8750 }
8751
8752 void bfsC(int c, int p){ //consider each permutation for all children
8753
8754     vector<int> tmp;
8755
8756     for(auto h:adj[c]){
8757         if(h!=p){
8758             bfsC(h,c);
8759             tmp.pb(h);
8760         }
8761     }
8762
8763     sort(all(tmp));
8764
8765     bfsp[c][0]=1;
8766
8767     if(tmp.empty()) return;
8768     ll z = tmp.size();
8769
8770     do{
8771         vector<int> y = lvl_cnt[c];
8772         rep_a(i,1,n) y[i]+=y[i-1];
8773         vector<int> y1;
8774
8775         for(auto h:tmp){
8776             y1 = lvl_cnt[h];
8777             rep_a(i,1,n) y1[i]+=y1[i-1];
8778             fun(y,y1, h,c,0);
8779
8780             rep(i,n) y[i]+=lvl_cnt[h][i];
8781         }
8782     }while(next_permutation(all(tmp)));
8783
8784     for(auto h:adj[c]){
8785         if(h!=p){
8786             fun1(h,c,fac_inv[z]);
8787         }
8788     }
8789 }
8790
8791
8792
8793
8794
8795

```

```

8796
8797
8798 }
8799
8800 //////////////////////////////////////////////////
8801
8802 void solve() {
8803     cin >> n;
8804
8805     adj.assign(n, vector<int>());
8806     lvl_cnt.assign(n, vector<int>(n,0));
8807     comp.assign(n, vector<ll>(n,0));
8808     subsz.assign(n,0);
8809
8810     dfsp.assign(n, vector<ll>(n,0));
8811     bfsp.assign(n, vector<ll>(n,0));
8812
8813     int x,y;
8814
8815     rep(i,n-1){
8816         cin>>x>>y;
8817         x--, y--;
8818
8819         adj[x].pb(y);
8820         adj[y].pb(x);
8821     }
8822
8823     cnt=0;
8824     chk(0,-1);
8825     assert(cnt==n);
8826
8827     dfsp[0][0]=1;
8828     dfsC(0,-1);
8829
8830     bfsC(0,-1);
8831
8832     ll ans = 0;
8833     rep(i,n){
8834         ll dz = 0;
8835         rep(j,n) dz+=dfsp[i][j];
8836         dz%=mod;
8837         rep(j,n){
8838             dz-=dfsp[i][j];
8839             dz%=mod;
8840             ans += (bfsp[i][j]*dz)%mod;
8841         }
8842         ans%=mod;
8843     }
8844
8845     if(ans<0) ans+=mod;
8846     cout<<ans<<'\n';
8847
8848
8849 }
8850
8851 signed main()
8852 {
8853
8854     #ifndef ONLINE_JUDGE
8855     freopen("input.txt", "r" , stdin);
8856     freopen("output.txt", "w" , stdout);
8857     #endif
8858     fast;
8859
8860     int t = 1;
8861     pre();
8862
8863     cin>>t;
8864

```

```

8865     for(int i=1;i<=t;i++)
8866     {
8867         solve();
8868     }
8869
8870     cerr << "Time : " << 1000 * ((double)clock()) / (double)CLOCKS_PER_SEC << "ms\n";
8871 }
8872
8873 //GOODINDICES
8874 #include <bits/stdc++.h>
8875 using namespace std;
8876
8877 /*
8878 -----Input Checker-----
8879 */
8880
8881 long long readInt(long long l,long long r,char endd){
8882     long long x=0;
8883     int cnt=0;
8884     int fi=-1;
8885     bool is_neg=false;
8886     while(true){
8887         char g=getchar();
8888         if(g=='-'){
8889             assert(fi==-1);
8890             is_neg=true;
8891             continue;
8892         }
8893         if('0'<=g && g<='9'){
8894             x*=10;
8895             x+=g-'0';
8896             if(cnt==0){
8897                 fi=g-'0';
8898             }
8899             cnt++;
8900             assert(fi!=0 || cnt==1);
8901             assert(fi!=0 || is_neg==false);
8902
8903             assert(!(cnt>19 || ( cnt==19 && fi>1) ));
8904         } else if(g==endd){
8905             if(is_neg){
8906                 x= -x;
8907             }
8908
8909             if(!(l <= x && x <= r))
8910             {
8911                 cerr << l << ' ' << r << ' ' << x << '\n';
8912                 assert(1 == 0);
8913             }
8914
8915             return x;
8916         } else {
8917             assert(false);
8918         }
8919     }
8920 }
8921
8922 string readString(int l,int r,char endd){
8923     string ret="";
8924     int cnt=0;
8925     while(true){
8926         char g=getchar();
8927         assert(g!=-1);
8928         if(g==endd){
8929             break;
8930         }
8931         cnt++;
8932         ret+=g;
8933     }

```

```

8934     assert(l<=cnt && cnt<=r);
8935     return ret;
8936 }
8937 long long readIntSp(long long l,long long r){
8938     return readInt(l,r,' ');
8939 }
8940 long long readIntLn(long long l,long long r){
8941     return readInt(l,r,'\n');
8942 }
8943 string readStringLn(int l,int r){
8944     return readString(l,r,'\n');
8945 }
8946 string readStringSp(int l,int r){
8947     return readString(l,r,' ');
8948 }
8949
8950
8951 /*
8952 -----Main code starts here-----
8953 */
8954
8955 const int MAX_T = 1e5;
8956 const int MAX_N = 1e5;
8957 const int MAX_SUM_LEN = 1e5;
8958
8959 #define fast ios_base::sync_with_stdio(0); cin.tie(0); cout.tie(0)
8960 #define ff first
8961 #define ss second
8962 #define mp make_pair
8963 #define ll long long
8964 #define rep(i,n) for(int i=0;i<n;i++)
8965 #define rev(i,n) for(int i=n;i>=0;i--)
8966 #define rep_a(i,a,n) for(int i=a;i<n;i++)
8967 #define pb push_back
8968
8969 int sum_n = 0, sum_m = 0;
8970 int max_n = 0, max_m = 0;
8971 int yess = 0;
8972 int nos = 0;
8973 int total_ops = 0;
8974 ll mod = 998244353;
8975
8976
8977 void solve(){
8978     int n = readIntLn(2,1000);
8979
8980     int freq[1001] = {0};
8981     int x;
8982     vector<int> a;
8983
8984     rep(i,n){
8985         if(i<n-1) x = readIntSp(1,1000);
8986         else x = readIntLn(1,1000);
8987
8988         a.pb(x);
8989         freq[x]++;
8990     }
8991
8992     sort(a.begin(), a.end());
8993
8994     int ans = 0;
8995     rep(i,1001) if(freq[i]>1) ans+=freq[i];
8996     bool z[1001] = {0};
8997
8998     rep_a(d,1,1000){
8999         int tmp = 0;
9000         for(auto i:a){
9001             if(z[i]) continue;
9002             int cnt = 0, sum = 0, grt1 = 0;

```

```

9003         for(int j=i; j<=1000+d; j+=d){
9004             if(j<=1000 && freq[j]) z[j]=1;
9005             if(j>1000 || freq[j]==0){
9006                 if(cnt%2==1 && !grt1) tmp += sum-1;
9007                 else tmp += sum;
9008
9009                 cnt=sum=grt1=0;
9010             }
9011             else{
9012                 cnt++;
9013                 sum+=freq[j];
9014                 if(freq[j]>1) grt1=1;
9015             }
9016         }
9017     }
9018     for(auto i:a) z[i] = 0;
9019
9020     ans = max(ans, tmp);
9021 }
9022
9023     cout<<ans<<'\\n';
9024 }
9025
9026 signed main()
9027 {
9028
9029     #ifndef ONLINE_JUDGE
9030     freopen("input.txt", "r" , stdin);
9031     freopen("output.txt", "w" , stdout);
9032     #endif
9033     fast;
9034
9035     int t = 1;
9036
9037     t = readIntLn(1,500);
9038
9039     for(int i=1;i<=t;i++)
9040     {
9041         solve();
9042     }
9043
9044     assert(getchar() == -1);
9045     assert(sum_n<=2000);
9046
9047     cerr << "Time : " << 1000 * ((double)clock()) / (double)CLOCKS_PER_SEC << "ms\\n";
9048 }
9049
9050 //SPLSTR
9051 using namespace std;
9052
9053 #define fo(i,n) for(i=0;i<n;i++)
9054 #define int long long
9055 #define deb(x) cout << #x << "=" << x << endl
9056 #define deb2(x, y) cout << #x << "=" << x << "," << #y << "=" << y << endl
9057 #define deb3(x, y, z) cout << #x << "=" << x << "," << #y << "=" << y << "," << #z <<
9058     "=" << z << endl
9059 #define pb push_back
9060 #define mp make_pair
9061 #define all(x) x.begin(), x.end()
9062 #define clr(x) memset(x, 0, sizeof(x))
9063 #define sortall(x) sort(all(x))
9064 #define el cout<<"\\n"
9065 #define max3(a,b,c) max(max((a), (b)), (c))
9066 #define max4(a,b,c,d) max(max((a), (b)),max((c), (d)))
9067 #define min3(a,b,c) min(min((a), (b)), (c))
9068 #define min4(a,b,c,d) min(min((a), (b)),min((c), (d)))
9069 ///////////////////////////////////////////////////
9070 int dx[] = {0, 0, -1, 1, 1, 1, -1, -1};
9071 int dy[] = {1, -1, 0, 0, -1, 1, 1, -1};

```

```

9071
9072 ////////////////for vectors
9073 # define maxv(a) (*max_element(a.begin(),a.end()))
9074 # define minv(a) (*min_element(a.begin(),a.end()))
9075 # define sumvi(a) (accumulate(a.begin(),a.end(),0LL))
9076 # define sumvd(a) (accumulate(a.begin(),a.end(),double(0)))
9077
9078 # define printv(v) {auto i = v;for(auto j : i) cout<< j << ' ';cout << "\n";}
9079 # define printvv(v) {auto i = v;for(auto j : i) {for(auto k : j) cout<< k << ' ';cout
<< "\n";}}
9080 # define prints(s) {auto i = s;for(auto j : i) cout<< j << ' ';cout << "\n";}
9081 # define printm(m) {auto i = m;for(auto j : i) cout<< j.first << ':' << j.second << '
';cout << "\n";}
9082
9083 ////////////////
9084 typedef pair<int, int> pii;
9085 typedef vector<int> vi;
9086 typedef vector<pii> vpii;
9087 typedef vector<vi> vvi;
9088 ////////////////
9089 mt1937_64 rang(chrono::high_resolution_clock::now().time_since_epoch().count());
9090 int rng(int lim) {
9091     uniform_int_distribution<int> uid(0, lim - 1);
9092     return uid(rang);
9093 }
9094 ////////////////
9095 const int inf = 1e9;
9096 const int INF = 1e18;
9097 const int mod = 1000000007;
9098 // const int mod = 998244353;
9099 const int N = 3e5 + 5, M = N;
9100 ////////////////
9101 int total = 0;
9102 void solve() {
9103     int i, j, n, k;
9104     cin >> n;
9105     total += n;
9106     assert( n >= 1 && n <= 200000);
9107     cin >> k;
9108     assert( n >= 1 && n <= mod - 7);
9109     string str;
9110     cin >> str;
9111     fo(i, n) {
9112         assert(str[i] == '0' || str[i] == '1');
9113     }
9114     int ones = 0;
9115     int zeros = 0;
9116     fo(i, n) {
9117         if (str[i] == '1') ones++;
9118         else zeros++;
9119     }
9120     int ans = abs( ones - zeros);
9121     ans += k;
9122     ans -= 1;
9123     ans /= k;
9124     cout << ans << "\n";
9125 }
9126
9127 int32_t main() {
9128     ios_base::sync_with_stdio(0), cin.tie(0), cout.tie(0);
9129     srand(chrono::high_resolution_clock::now().time_since_epoch().count());
9130     int t = 1;
9131     cin >> t;
9132     while (t--) {
9133         solve();
9134     }
9135     assert(total <= 200000);
9136     return 0;
9137 }

```

```

9138
9139 //SPLSTR-EDITOR
9140 using namespace std;
9141
9142 int main() {
9143     int T;
9144     cin >> T;
9145     while(T--){
9146         int n,k;
9147         cin >> n >> k;
9148         string s;
9149         cin >> s;
9150         int x=0;
9151         for(int i=0;i<n;i++){
9152             if(s[i]=='0')x++;
9153             else x--;
9154         }
9155         x=abs(x);
9156         cout << x/k + (int)(x%k!=0) << endl;
9157     }
9158     return 0;
9159 }
9160
9161 //CIREQ
9162 using namespace std;
9163
9164 int main() {
9165     ios_base :: sync_with_stdio(0);
9166     cin.tie(0); cout.tie(0);
9167     int t;
9168     cin >> t;
9169     while (t--) {
9170         int n;
9171         cin >> n;
9172         vector<int> a(n + 1);
9173         for (int i = 0; i < n; i++) {
9174             int x;
9175             cin >> x;
9176             a[x]++;
9177         }
9178         int sum = 0, ans = 0;
9179         for (int i = 1; i <= n; i++) {
9180             sum += a[i];
9181             ans = max(ans, (sum + i - 1) / i);
9182         }
9183         cout << ans << '\n';
9184     }
9185     return 0;
9186 }
9187
9188
9189 //CIREQ-EDITOR
9190 using namespace std;
9191
9192 int main() {
9193     int T;
9194     cin >> T;
9195     while(T--){
9196         int n;
9197         cin >> n;
9198         vector<int>v(n);
9199         for(int i=0;i<n;i++)cin >> v[i];
9200         sort(v.begin(),v.end());
9201         priority_queue<int,vector<int>,greater<int>>pq;
9202         pq.push(1);
9203         for(int i=0;i<n;i++){
9204             if(v[i]>=pq.top()){
9205                 pq.push(pq.top()+1);
9206                 pq.pop();

```

```

9207         }
9208         else{
9209             pq.push(2);
9210         }
9211     }
9212     cout << pq.size() << endl;
9213 }
9214 return 0;
9215 }
9216
9217 //STRPERM
9218 using namespace std;
9219
9220 void solve(int tc) {
9221     int n, m;
9222     cin >> n >> m;
9223     vector<int> last(n + 1, n);
9224     vector<vector<int>> v(n + 1);
9225     for (int i = 1; i <= m; i++) {
9226         int x, y;
9227         cin >> x >> y;
9228         last[x] = y;
9229     }
9230     for (int i = 1; i <= n; i++) {
9231         v[last[i]].push_back(i);
9232     }
9233     vector<int> ans(n + 1);
9234     priority_queue<int> pq;
9235     for (int i = n; i >= 1; i--) {
9236         for (auto &u : v[i]) {
9237             pq.push(u);
9238         }
9239         if (pq.empty()) {
9240             cout << "-1\n";
9241             return;
9242         }
9243         ans[i] = pq.top();
9244         pq.pop();
9245     }
9246     for (int i = 1; i <= n; i++) {
9247         cout << ans[i] << ' ';
9248     }
9249     cout << '\n';
9250 }
9251
9252 int main() {
9253     ios_base :: sync_with_stdio(0);
9254     cin.tie(0);
9255     int t = 1;
9256     cin >> t;
9257     for (int i = 1; i <= t; i++) solve(i);
9258     return 0;
9259 }
9260
9261 //STRPERM-EDITOR
9262 using namespace std;
9263
9264 int main() {
9265     int T;
9266     cin >> T;
9267     while(T--){
9268         int n,m;
9269         cin >> n >> m;
9270         vector<int>v[n+1];
9271         unordered_map<int,int>ff;
9272         for(int i=0;i<m;i++){
9273             int x,y;
9274             cin >> x >> y;

```



```

9276         v[y].push_back(x);
9277         ff[x]++;
9278     }
9279     for(int i=1;i<=n;i++){
9280         if(ff[i])continue;
9281         v[n].push_back(i);
9282     }
9283     vector<int>ans;
9284     priority_queue<int>pq;
9285     for(int i=n;i>=1;i--){
9286         for(auto j:v[i])pq.push(j);
9287         if(pq.empty()){
9288             cout << -1 << endl;
9289             break;
9290         }
9291         ans.push_back(pq.top());
9292         pq.pop();
9293     }
9294     if(ans.size()<n)continue;
9295     for(int i=0;i<n;i++)cout << ans[n-i-1] << " ";
9296     cout << endl;
9297 }
9298 return 0;
9299 }
9300
9301 //TREEEXOR
9302 using namespace std;
9303 const int N = 2e5 + 10;
9304
9305 int n, k, sum[N];
9306 vector<int> g[N];
9307
9308 string ans;
9309
9310 void dfs(int v) {
9311     for (auto u : g[v]) {
9312         dfs(u);
9313         sum[v] += sum[u];
9314     }
9315     if (k) {
9316         if (sum[v] % 2 == 0) {
9317             ans[v] = '1';
9318             sum[v]++;
9319         }
9320         k--;
9321     } else {
9322         if (sum[v] % 2) {
9323             ans[v] = '1';
9324             sum[v]++;
9325         }
9326     }
9327 }
9328 int main() {
9329     ios_base :: sync_with_stdio(0);
9330     cin.tie(0); cout.tie(0);
9331     int t;
9332     cin >> t;
9333     while (t--) {
9334         cin >> n >> k;
9335         ans.clear();
9336         for (int i = 0; i < n; i++) {
9337             g[i].clear();
9338             sum[i] = 0;
9339             ans.push_back('0');
9340         }
9341         for (int i = 1; i < n; i++) {
9342             int x;
9343             cin >> x;
9344             g[x - 1].push_back(i);

```

```

9345     }
9346     dfs(0);
9347     cout << ans << '\n';
9348 }
9349 return 0;
9350 }
9351
9352 //TREEEXOR-EDITOR
9353 using namespace std;
9354 int done;
9355 int dfs(vector<vector<int>>&adj,int x,vector<char>&ans,vector<bool>&vi){
9356     int y=0;
9357     for(auto j:adj[x]){
9358         if(vi[j])continue;
9359         vi[j]=true;
9360         y+=dfs(adj,j,ans,vi);
9361     }
9362     if(done){
9363         ans[x]='0'+1-(y%2);
9364         done--;
9365         return 1;
9366     }
9367     ans[x]='0'+(y%2);
9368     return 0;
9369 }
9370 int main() {
9371     int T;
9372     cin >> T;
9373     while(T--){
9374         int n,k;
9375         cin >> n >> k;
9376         vector<vector<int>>adj(n+1);
9377         for(int i=0;i<n-1;i++){
9378             int t;
9379             cin >> t;
9380             adj[i+2].push_back(t);
9381             adj[t].push_back(i+2);
9382         }
9383         done=k;
9384         vector<char>ans(n+1,'0');
9385         vector<bool>vi(n+1,false);
9386         vi[1]=true;
9387         int m=dfs(adj,1,ans,vi);
9388         for(int i=1;i<=n;i++)cout << ans[i];
9389         cout << endl;
9390     }
9391     return 0;
9392 }
9393
9394 //MEXCCHF
9395 #include<bits/stdc++.h>
9396 using namespace std;
9397
9398 const int mod = 1e9 + 7;
9399
9400 void add(int &a, int b) {
9401     a = (a + b) % mod;
9402 }
9403 int mul (int a, int b) {
9404     return 1LL * a * b % mod;
9405 }
9406
9407 int32_t main() {
9408     int n, k;
9409     cin >> n >> k;
9410     vector<vector<int>> dp(n + 1, vector<int>(k + 1));
9411     for (int i = 1; i <= n; i++) {
9412         dp[i][n - i + 1] = 1;
9413     }

```

```

9414     for (int i = 1; i < n; i++) {
9415         vector<vector<int>> ndp(n + 1, vector<int>(k + 1));
9416         for (int s = 1; s <= k; s++) {
9417             int sum = 0;
9418             for (int mx_pos = 1; mx_pos <= n; mx_pos++) {
9419                 int ns = s + n - mx_pos + 1;
9420                 // i is on new mx_pos
9421                 if (ns <= k) {
9422                     add(ndp[mx_pos][ns], sum);
9423                 }
9424                 // i is put on position < mx_pos (one of 0, 1, ..., i - 1 is on mx_pos)
9425                 if (ns <= k && mx_pos > i) {
9426                     add(ndp[mx_pos][ns], mul(mx_pos - i, dp[mx_pos][s]));
9427                 }
9428                 add(sum, dp[mx_pos][s]);
9429             }
9430         }
9431         dp = ndp;
9432     }
9433     for (int i = n; i <= k; i++) {
9434         cout << dp[n][i] << " \n"[i == k];
9435     }
9436     return 0;
9437 }
9438
9439 //FIXSUM
9440 #include<bits/stdc++.h>
9441 using namespace std;
9442 template<class T> inline T Bit(T x, int i) { return (x >> i) & 1;}
9443
9444 void solve(int tc) {
9445     int n, s;
9446     cin >> n >> s;
9447     int l = 1, r = s, ans;
9448     while (l <= r) {
9449         int mid = (l + r) / 2;
9450         long long sum = 0;
9451         int cnt = 0;
9452         for (int i = 30; i >= 0; i--) {
9453             if (Bit(mid, i)) {
9454                 sum += 1LL * (n - 1) * (1LL << i);
9455                 cnt = min(++cnt, n - 1);
9456             } else {
9457                 sum += 1LL * cnt * (1LL << i);
9458             }
9459         }
9460         if (sum >= s) {
9461             r = mid - 1;
9462             ans = mid;
9463         } else {
9464             l = mid + 1;
9465         }
9466     }
9467     cout << ans << '\n';
9468 }
9469
9470 int main() {
9471     ios_base :: sync_with_stdio(0); cin.tie(0); cout.tie(0);
9472     int t = 1;
9473     cin >> t;
9474     for (int i = 1; i <= t; i++) solve(i);
9475 }
9476
9477 //COPYPUSH
9478 #include "bits/stdc++.h"
9479 using namespace std;
9480 #define ll long long
9481 #define pb push_back
9482 #define all(_obj) _obj.begin(), _obj.end()

```

```

9483 #define F first
9484 #define S second
9485 #define pll pair<ll, ll>
9486 #define vll vector<ll>
9487 ll INF = 1e18;
9488 const int N = 1e5 + 11, mod = 1e9 + 7;
9489 ll max(ll a, ll b) { return ((a > b) ? a : b); }
9490 ll min(ll a, ll b) { return ((a > b) ? b : a); }
9491 mt19937 rng(chrono::steady_clock::now().time_since_epoch().count());
9492 void sol(void)
9493 {
9494     int n;
9495     cin >> n;
9496     string s;
9497     cin >> s;
9498     while (s.size())
9499     {
9500         if (s.size() % 2)
9501             s.pop_back();
9502         else
9503         {
9504             if (s.substr(0, s.size() / 2) != s.substr(s.size() / 2, s.size() / 2))
9505             {
9506                 cout << "NO\n";
9507                 return;
9508             }
9509             s = s.substr(0, s.size() / 2);
9510         }
9511     }
9512     cout << "YES\n";
9513     return;
9514 }
9515 int main()
9516 {
9517     ios_base::sync_with_stdio(false);
9518     cin.tie(NULL), cout.tie(NULL);
9519     int test = 1;
9520     cin >> test;
9521     while (test--)
9522         sol();
9523 }
9524
9525 //ENDSORTED
9526 //Utkarsh.25dec
9527 #include <bits/stdc++.h>
9528 #define ll long long int
9529 #define pb push_back
9530 #define mp make_pair
9531 #define mod 1000000007
9532 #define vl vector<ll>
9533 #define all(c) (c).begin(), (c).end()
9534 using namespace std;
9535 ll power(ll a, ll b) { ll res=1; a%=mod; assert(b>=0);
9536 for(; b>=1; b>>=1) { if(b&1) res=res*a%mod; a=a*a%mod; } return res; }
9537 ll modInverse(ll a) { return power(a, mod-2); }
9538 const int N=500023;
9539 bool vis[N];
9540 vector<int> adj[N];
9541 long long readInt(long long l, long long r, char endd) {
9542     long long x=0;
9543     int cnt=0;
9544     int fi=-1;
9545     bool is_neg=false;
9546     while(true) {
9547         char g=getchar();
9548         if(g=='-') {
9549             assert(fi==-1);
9550             is_neg=true;
9551             continue;

```

```

9551     }
9552     if('0'<=g && g<='9'){
9553         x*=10;
9554         x+=g-'0';
9555         if(cnt==0){
9556             fi=g-'0';
9557         }
9558         cnt++;
9559         assert(fi!=0 || cnt==1);
9560         assert(fi!=0 || is_neg==false);
9561
9562         assert(!(cnt>19 || ( cnt==19 && fi>1) ));
9563     } else if(g==endd){
9564         if(is_neg){
9565             x= -x;
9566         }
9567
9568         if(!(l <= x && x <= r))
9569         {
9570             cerr << l << ' ' << r << ' ' << x << '\n';
9571             assert(1 == 0);
9572         }
9573
9574         return x;
9575     } else {
9576         assert(false);
9577     }
9578 }
9579
9580 string readString(int l,int r,char endd){
9581     string ret="";
9582     int cnt=0;
9583     while(true){
9584         char g=getchar();
9585         assert(g!=-1);
9586         if(g==endd){
9587             break;
9588         }
9589         cnt++;
9590         ret+=g;
9591     }
9592     assert(l<=cnt && cnt<=r);
9593     return ret;
9594 }
9595 long long readIntSp(long long l,long long r){
9596     return readInt(l,r,' ');
9597 }
9598 long long readIntLn(long long l,long long r){
9599     return readInt(l,r,'\n');
9600 }
9601 string readStringLn(int l,int r){
9602     return readString(l,r,'\n');
9603 }
9604 string readStringSp(int l,int r){
9605     return readString(l,r,' ');
9606 }
9607 int sumN=0;
9608 void solve()
9609 {
9610     int n=readInt(2,100000,'\n');
9611     sumN+=n;
9612     assert(sumN<=300000);
9613     int P[n+1]={0};
9614     int indx[n+1]={0};
9615     int mark[n+1]={0};
9616     for(int i=1;i<=n;i++)
9617     {
9618         if(i!=n)
9619             P[i]=readInt(1,n,' ');

```

```

9620         else
9621             P[i]=readInt(1,n,'\n');
9622             indx[P[i]]=i;
9623             mark[P[i]]=1;
9624     }
9625     for(int i=1;i<=n;i++)
9626         assert(mark[i]==1);
9627     int a=indx[1];
9628     int b=indx[n];
9629     int ans=(a-1)+(n-b);
9630     if(a>b)
9631         ans--;
9632     cout<<ans<<'\n';
9633 }
9634 int main()
9635 {
9636     #ifndef ONLINE_JUDGE
9637     freopen("input.txt", "r", stdin);
9638     freopen("output.txt", "w", stdout);
9639     #endif
9640     ios_base::sync_with_stdio(false);
9641     cin.tie(NULL),cout.tie(NULL);
9642     int T=readInt(1,1000,'\n');
9643     while(T--)
9644         solve();
9645     assert(getchar()==-1);
9646     cerr << "Time : " << 1000 * ((double)clock()) / (double)CLOCKS_PER_SEC << "ms\n";
9647 }
9648
9649 //ENDSORTED-EDITOR
9650 #include "bits/stdc++.h"
9651 using namespace std;
9652 #define ll long long
9653 #define pb push_back
9654 #define all(_obj) _obj.begin(), _obj.end()
9655 #define F first
9656 #define S second
9657 #define pll pair<ll, ll>
9658 #define vll vector<ll>
9659 ll INF = 1e18;
9660 const int N = 1e5 + 11, mod = 1e9 + 7;
9661 ll max(ll a, ll b) { return ((a > b) ? a : b); }
9662 ll min(ll a, ll b) { return ((a > b) ? b : a); }
9663 mt19937 rng(chrono::steady_clock::now().time_since_epoch().count());
9664 void sol(void)
9665 {
9666     int n, id1, idn;
9667     cin >> n;
9668     vll p(n);
9669     for (int i = 0; i < n; i++)
9670     {
9671         cin >> p[i];
9672         if (p[i] == 1)
9673             id1 = i;
9674         else if (p[i] == n)
9675             idn = i;
9676     }
9677     if (id1 < idn)
9678         cout << id1 + n - 1 - idn << '\n';
9679     else
9680         cout << id1 + n - 1 - idn - 1 << '\n';
9681
9682     return;
9683 }
9684 int main()
9685 {
9686     ios_base::sync_with_stdio(false);
9687     cin.tie(NULL), cout.tie(NULL);
9688     int test = 1;

```

```

9689     cin >> test;
9690     while (test--)
9691         sol();
9692 }
9693
9694
9695 //ORXOR
9696
9697 #include <bits/stdc++.h>
9698 using namespace std;
9699
9700 int main() {
9701     int t;
9702     cin >> t;
9703     while(t--) {
9704         int n, x;
9705         cin >> n >> x;
9706         int bk = 0;
9707         for(int i = 20; i > 0; i--) {
9708             if(n&(1LL<<i)) {
9709                 bk = (1LL<<(i + 1));
9710                 break;
9711             }
9712         }
9713         if((n == 2 && x != 3) || ((n&(n - 1)) == 0 && (x&n) == 0) || x >= bk) {
9714             cout << "-1\n";
9715             continue;
9716         }
9717         if(n == 2) {
9718             cout << "1 1 2\n";
9719             continue;
9720         }
9721         set<int> s1, s2;
9722         for(int i = 1; i <= n; i++) s1.insert(i);
9723         for(int i = 0; i < 20; i++) {
9724             if((1LL<<i) > n) break;
9725             if((x&(1LL<<i)) == 0) {
9726                 s1.erase((1LL<<i));
9727                 s2.insert((1LL<<i));
9728             }
9729         }
9730         int n1 = 0, n2 = 0;
9731         if(s1.size()) {
9732             n1 = (*(s1.begin()));
9733             s1.erase(s1.begin());
9734             while(s1.size()) {
9735                 int now = (*(s1.begin()));
9736                 cout << "1 " << n1 << " " << now << "\n";
9737                 n1 |= now;
9738                 s1.erase(s1.begin());
9739             }
9740         }
9741         if(s2.size()) {
9742             n2 = (*(s2.begin()));
9743             s2.erase(s2.begin());
9744             while(s2.size()) {
9745                 int now = (*(s2.begin()));
9746                 cout << "1 " << n2 << " " << now << "\n";
9747                 n2 |= now;
9748                 s2.erase(s2.begin());
9749             }
9750         }
9751         if(n1 > 0 && n2 > 0) cout << "2 " << n1 << " " << n2 << "\n";
9752     }
9753 }
9754
9755 //ORXOR - EDITOR
9756 #include "bits/stdc++.h"
9757 using namespace std;

```

```

9758 #define ll long long
9759 #define pb push_back
9760 #define all(_obj) _obj.begin(), _obj.end()
9761 #define F first
9762 #define S second
9763 #define pll pair<ll, ll>
9764 #define vll vector<ll>
9765 ll INF = 1e18;
9766 const int N = 1e5 + 11, mod = 1e9 + 7;
9767 ll max(ll a, ll b) { return ((a > b) ? a : b); }
9768 ll min(ll a, ll b) { return ((a > b) ? b : a); }
9769 mt19937 rng(chrono::steady_clock::now().time_since_epoch().count());
9770 void sol(void)
9771 {
9772     int n, x, msbx, msbn;
9773     cin >> n >> x;
9774     map<int, bool> ispow2;
9775     if (n == 2 && x != 3)
9776     {
9777         cout << -1 << '\n';
9778         return;
9779     }
9780     else if(n==2 && x==3)
9781     {
9782         cout<<1<<' '<<1<<' '<<2<<'\n';
9783         return ;
9784     }
9785     for (int i = 0; i < 20; i++)
9786     {
9787         ispow2[1 << i] = true;
9788         if ((1 << i) & n)
9789             msbn = i;
9790         if ((1 << i) & x)
9791             msbx = i;
9792     }
9793     if (msbx > msbn)
9794     {
9795         cout << -1 << '\n';
9796         return;
9797     }
9798     if (!ispow2[n] || (x & n))
9799     {
9800         int val = 0;
9801         for (int i = 1; i <= n; i++)
9802         {
9803             if (!ispow2[i])
9804             {
9805                 if (i > 3)
9806                     cout << 1 << ' ' << i << ' ' << val << '\n';
9807                 val |= i;
9808             }
9809         }
9810         for (auto y : ispow2)
9811         {
9812             if (y.F > n || !y.S)
9813                 continue;
9814             if (x & (y.F))
9815             {
9816                 cout << 1 << ' ' << y.F << ' ' << val << '\n';
9817                 val |= y.F;
9818             }
9819             else
9820             {
9821                 cout << 2 << ' ' << y.F << ' ' << val << '\n';
9822                 val ^= y.F;
9823             }
9824         }
9825     }
9826     else

```



```

9827     {
9828         cout << -1 << '\n';
9829     }
9830
9831     return;
9832 }
9833 int main()
9834 {
9835     ios_base::sync_with_stdio(false);
9836     cin.tie(NULL), cout.tie(NULL);
9837     int test = 1;
9838     cin >> test;
9839     while (test--)
9840         sol();
9841 }
9842
9843 //HEIGHTS
9844 #include <bits/stdc++.h>
9845 using namespace std;
9846
9847 int main() {
9848     //freopen("inp7.in", "r", stdin);
9849     //freopen("out7.out", "w", stdout);
9850     int t;
9851     cin >> t;
9852     while(t--) {
9853         int n;
9854         cin >> n;
9855         map<int, int> cnt;
9856         int a[n], mx = 0;
9857         for(int i = 0; i < n; i++) cin >> a[i], mx = max(mx, a[i]), cnt[a[i]]++;
9858         int bad = 0, g2 = 0, largest = 0;
9859         for(int i = 0; i < n; i++) {
9860             if(cnt[a[i]] == 1) {
9861                 bad++;
9862                 if(mx == a[i]) largest = 1;
9863             }
9864             else if(cnt[a[i]] > 2) g2++;
9865         }
9866         if(bad == 1) {
9867             if(g2 || !largest) cout << 1 << "\n";
9868             else cout << 2 << "\n";
9869         } else cout << (bad + 1)/2 << "\n";
9870     }
9871 }
9872
9873 //HEIGHTS-EDITOR
9874 #include "bits/stdc++.h"
9875 using namespace std;
9876 #define ll long long
9877 #define pb push_back
9878 #define all(_obj) _obj.begin(), _obj.end()
9879 #define F first
9880 #define S second
9881 #define pll pair<ll, ll>
9882 #define vll vector<ll>
9883 ll INF = 1e18;
9884 const int N = 1e5 + 11, mod = 1e9 + 7;
9885 ll max(ll a, ll b) { return ((a > b) ? a : b); }
9886 ll min(ll a, ll b) { return ((a > b) ? b : a); }
9887 mt19937 rng(chrono::steady_clock::now().time_since_epoch().count());
9888 void sol(void)
9889 {
9890     int n, cnt = 0, gp3 = 0;
9891     cin >> n;
9892     map<int, int> mp;
9893     vll v(n);
9894     for (int i = 0; i < n; i++)
9895     {

```

```

9896         cin >> v[i];
9897         mp[v[i]]++;
9898     }
9899     sort(all(v));
9900     for (auto x : mp)
9901     {
9902         if (x.S == 1)
9903             cnt++;
9904         if (x.S > 2)
9905             gp3++;
9906     }
9907     if (cnt ==1 && !gp3 && mp[v.back()]==1)
9908     {
9909         cout<<2<<'\n';
9910         return ;
9911     }
9912     cout << (cnt + 1) / 2 << '\n';
9913     return;
9914 }
9915 int main()
9916 {
9917     ios_base::sync_with_stdio(false);
9918     cin.tie(NULL), cout.tie(NULL);
9919     int test = 1;
9920     cin >> test;
9921     while (test--)
9922         sol();
9923 }
9924
9925 //MINCOST
9926 #include <bits/stdc++.h>
9927 using namespace std;
9928 typedef long long llo;
9929 #define a first
9930 #define b second
9931 #define pb push_back
9932 #define endl '\n'
9933
9934
9935
9936 void setIO(string name) {
9937     ios_base::sync_with_stdio(0); cin.tie(0);
9938     freopen((name+".in").c_str(),"r",stdin);
9939
9940     freopen((name+".out").c_str(),"w",stdout);
9941 }
9942 vector<llo> adj[200001];
9943 llo dp[200001][2];
9944 llo aa[200001];
9945 llo ll[200001];
9946 llo rr[200001];
9947 llo cc[200001];
9948 void dfs(llo no,llo par=-1){
9949     dp[no][0]=0;
9950     dp[no][1]=0;
9951     for(auto j:adj[no]){
9952         if(j!=par){
9953             dfs(j,no);
9954             dp[no][1]+=dp[j][0]+abs(aa[j]-cc[no]);
9955             dp[no][0]+=min(dp[j][1]+abs(aa[no]-cc[j]),dp[j][0]+abs(aa[no]-aa[j]));
9956         }
9957     }
9958 }
9959 }
9960 int main(){
9961
9962     //for(int ii=1;ii<=12;ii++){
9963         //setIO("12");
9964         llo t;

```

```

9965         cin>>t;
9966         while(t--){
9967             llo n;
9968             cin>>n;
9969             for(llo i=0;i<n;i++){
9970                 adj[i].clear();
9971             }
9972             llo zz=5;
9973             for(llo i=0;i<n;i++){
9974                 cin>>ll[i]>>aa[i]>>rr[i];
9975             }
9976             for(llo i=0;i<n-1;i++){
9977                 llo aa,bb;
9978                 cin>>aa>>bb;
9979                 aa--;
9980                 bb--;
9981                 adj[aa].pb(bb);
9982                 adj[bb].pb(aa);
9983             }
9984             for(llo i=0;i<n;i++){
9985                 vector<llo> cur;
9986                 for(auto j:adj[i]){
9987                     cur.pb(aa[j]);
9988                 }
9989                 sort(cur.begin(),cur.end());
9990                 llo x=cur.size();
9991                 cc[i]=cur[x/2];
9992                 if(cc[i]<ll[i]){
9993                     cc[i]=ll[i];
9994                 }
9995                 else if(cc[i]>rr[i]){
9996                     cc[i]=rr[i];
9997                 }
9998             }
9999             dfs(0);
10000             cout<<min(dp[0][0],dp[0][1])<<endl;
10001         }
10002     //}
10003
10004     return 0;
10005 }
10006
10007 #include "bits/stdc++.h"
10008 using namespace std;
10009 #define ll long long
10010 #define pb push_back
10011 #define all(_obj) _obj.begin(), _obj.end()
10012 #define F first
10013 #define S second
10014 #define pll pair<ll, ll>
10015 #define vll vector<ll>
10016 ll INF = 1e18;
10017 const int N = 2e5 + 11, mod = 1e9 + 7;
10018 ll max(ll a, ll b) { return ((a > b) ? a : b); }
10019 ll min(ll a, ll b) { return ((a > b) ? b : a); }
10020 mt19937 rng(chrono::steady_clock::now().time_since_epoch().count());
10021 int L[N], R[N], A[N];
10022 vll v[N];
10023 ll dp[N][2], optimal[N];
10024 void dfs(int u, int p = -1)
10025 {
10026     dp[u][0] = 0;
10027     dp[u][1] = 0;
10028     for (auto x : v[u])
10029     {
10030         if (x != p)
10031         {
10032             dfs(x, u);
10033             dp[u][1] += dp[x][0] + abs(A[x] - optimal[u]);

```

```

10034         dp[u][0] += min(dp[x][1] + abs(A[u] - optimal[x]), dp[x][0] + abs(A[u] -
10035             A[x]));
10036     }
10037 }
10038 void sol(void)
10039 {
10040     int n;
10041     cin >> n;
10042     for (int i = 1; i <= n; i++)
10043         cin >> L[i] >> A[i] >> R[i], v[i].clear();
10044     for (int i = 0; i < n - 1; i++)
10045     {
10046         int a, b;
10047         cin >> a >> b;
10048         v[a].pb(b);
10049         v[b].pb(a);
10050     }
10051     for (int i = 1; i <= n; i++)
10052     {
10053         vll cur;
10054         for (auto x : v[i])
10055         {
10056             cur.pb(A[x]);
10057         }
10058         sort(all(cur));
10059         optimal[i] = cur[cur.size() / 2];
10060         if (optimal[i] < L[i])
10061         {
10062             optimal[i] = L[i];
10063         }
10064         else if (optimal[i] > R[i])
10065         {
10066             optimal[i] = R[i];
10067         }
10068     }
10069     dfs(1);
10070     cout << min(dp[1][0], dp[1][1]) << endl;
10071     return;
10072 }
10073 int main()
10074 {
10075     ios_base::sync_with_stdio(false);
10076     cin.tie(NULL), cout.tie(NULL);
10077     int test = 1;
10078     cin >> test;
10079     while (test--)
10080         sol();
10081 }
10082
10083 //TOOMANYLIS2
10084
10085 #include<bits/stdc++.h>
10086 using namespace std;
10087 #define ll long long int
10088 #define pb push_back
10089 #define rb pop_back
10090 #define ti tuple<int, int, int>
10091 #define pii pair<int, int>
10092 #define pli pair<ll, int>
10093 #define pll pair<ll, ll>
10094 #define mp make_pair
10095 #define mt make_tuple
10096
10097 using namespace std;
10098
10099 FILE *fp;
10100 ofstream outfile;
10101

```

```

10102 long long readInt(long long l,long long r,char endd){
10103     long long x=0;
10104     int cnt=0;
10105     int fi=-1;
10106     bool is_neg=false;
10107     while(true){
10108         char g=getchar();
10109         if(g=='-'){
10110             assert(fi==-1);
10111             is_neg=true;
10112             continue;
10113         }
10114         if('0'<=g && g<='9'){
10115             x*=10;
10116             x+=g-'0';
10117             if(cnt==0){
10118                 fi=g-'0';
10119             }
10120             cnt++;
10121             assert(fi!=0 || cnt==1);
10122             assert(fi!=0 || is_neg==false);
10123
10124             assert(!(cnt>19 || ( cnt==19 && fi>1) ));
10125         } else if(g==endd){
10126             if(is_neg){
10127                 x= -x;
10128             }
10129             assert(l<=x && x<=r);
10130             return x;
10131         } else {
10132             assert(false);
10133         }
10134     }
10135 }
10136 string readString(int l,int r,char endd){
10137     string ret="";
10138     int cnt=0;
10139     while(true){
10140         char g=getchar();
10141         assert(g != -1);
10142         if(g==endd){
10143             break;
10144         }
10145         cnt++;
10146         ret+=g;
10147     }
10148     assert(l<=cnt && cnt<=r);
10149     return ret;
10150 }
10151 long long readIntSp(long long l,long long r){
10152     return readInt(l,r,' ');
10153 }
10154 long long readIntLn(long long l,long long r){
10155     return readInt(l,r,'\n');
10156 }
10157 string readStringLn(int l,int r){
10158     return readString(l,r,'\n');
10159 }
10160 string readStringSp(int l,int r){
10161     return readString(l,r,' ');
10162 }
10163
10164 const string newln = "\n", space = " ";
10165 const int maxt = 500, maxn = 200000, maxsumn = 200000, mod = 1e9 + 7, MAX = 200010;
10166 ll fac[MAX], ifac[MAX];
10167 ll rpe(ll a, int b){
10168     ll ans = 1;
10169     while(b != 0){
10170         if(b & 1)ans = ans * a % mod;

```

```

10171         a = a * a % mod; b >>= 1;
10172     }
10173     return ans;
10174 }
10175 ll ncr(int n, int d){
10176     if(d > n || d < 0) return 0;
10177     return fac[n] * ifac[d] % mod * ifac[n - d] % mod;
10178 }
10179 ll cal(int n, int i){
10180     if(i < n / 2) return 0;
10181     return (ncr(n, n - i) - ncr(n, n - i - 1) + mod) % mod;
10182 }
10183 int main()
10184 {
10185     fac[0] = 1;
10186     for(int i = 1; i < MAX; i++){
10187         fac[i] = i * fac[i - 1] % mod;
10188     }
10189     ifac[MAX - 1] = rpe(fac[MAX - 1], mod - 2);
10190     for(int i = MAX - 2; i >= 0; i--){
10191         ifac[i] = (i + 1) * ifac[i + 1] % mod;
10192     }
10193     int sumn = 0;
10194     int t = readIntLn(1, maxt);
10195     while(t--){
10196         int n = readIntLn(1, maxn); sumn += n;
10197         for(int i = 1; i <= n; i++) cout << cal(n, i) << " \n"[i == n];
10198     }
10199     assert(sumn <= maxsumn);
10200     assert(getchar() == -1);
10201 }
10202
10203 //ARRSORT
10204 #include <bits/stdc++.h>
10205 using namespace std;
10206
10207
10208 /*
10209 -----Input Checker-----
10210 */
10211
10212 long long readInt(long long l, long long r, char endd){
10213     long long x = 0;
10214     int cnt = 0;
10215     int fi = -1;
10216     bool is_neg = false;
10217     while(true){
10218         char g = getchar();
10219         if(g == '-'){
10220             assert(fi == -1);
10221             is_neg = true;
10222             continue;
10223         }
10224         if('0' <= g && g <= '9'){
10225             x *= 10;
10226             x += g - '0';
10227             if(cnt == 0){
10228                 fi = g - '0';
10229             }
10230             cnt++;
10231             assert(fi != 0 || cnt == 1);
10232             assert(fi != 0 || is_neg == false);
10233
10234             assert(!(cnt > 19 || (cnt == 19 && fi > 1)));
10235         } else if(g == endd){
10236             if(is_neg){
10237                 x = -x;
10238             }
10239         }

```

```

10240         if(!(l <= x && x <= r))
10241         {
10242             cerr << l << ' ' << r << ' ' << x << '\n';
10243             assert(1 == 0);
10244         }
10245
10246         return x;
10247     } else {
10248         assert(false);
10249     }
10250 }
10251 }
10252 string readString(int l,int r,char endd){
10253     string ret="";
10254     int cnt=0;
10255     while(true){
10256         char g=getchar();
10257         assert(g!=-1);
10258         if(g==endd){
10259             break;
10260         }
10261         cnt++;
10262         ret+=g;
10263     }
10264     assert(l<=cnt && cnt<=r);
10265     return ret;
10266 }
10267 long long readIntSp(long long l,long long r){
10268     return readInt(l,r,' ');
10269 }
10270 long long readIntLn(long long l,long long r){
10271     return readInt(l,r,'\n');
10272 }
10273 string readStringLn(int l,int r){
10274     return readString(l,r,'\n');
10275 }
10276 string readStringSp(int l,int r){
10277     return readString(l,r,' ');
10278 }
10279
10280
10281 /*
10282 -----Main code starts here-----
10283 */
10284
10285 const int MAX_T = 1e5;
10286 const int MAX_N = 1e5;
10287 const int MAX_SUM_LEN = 1e5;
10288
10289 #define fast ios_base::sync_with_stdio(0); cin.tie(0); cout.tie(0)
10290 #define ff first
10291 #define ss second
10292 #define mp make_pair
10293 #define ll long long
10294 #define rep(i,n) for(int i=0;i<n;i++)
10295 #define rev(i,n) for(int i=n;i>=0;i--)
10296 #define rep_a(i,a,n) for(int i=a;i<n;i++)
10297 #define pb push_back
10298
10299 ll sum_n = 0, sum_m = 0;
10300 int max_n = 0, max_m = 0;
10301 int yess = 0;
10302 int nos = 0;
10303 int total_ops = 0;
10304 ll mod = 998244353;
10305
10306 using ii = pair<ll,ll>;
10307
10308

```

```

10309
10310 void solve() {
10311     int n = readIntLn(2,1e5);
10312     sum_n+=n;
10313
10314     int p[n];
10315     rep(i,n){
10316         if(i<n-1) p[i] = readIntSp(1,n);
10317         else p[i] = readIntLn(1,n);
10318     }
10319
10320     int g = 0;
10321     rep(i,n){
10322         g = __gcd(g, abs((p[i]-1)-i));
10323     }
10324
10325
10326     sort(p,p+n);
10327     rep(i,n) assert(p[i]==i+1);
10328
10329     if(g==0) g=-1;
10330     cout<<g<<'\n';
10331 }
10332
10333
10334
10335 signed main()
10336 {
10337
10338     #ifndef ONLINE_JUDGE
10339     freopen("input.txt", "r" , stdin);
10340     freopen("output.txt", "w" , stdout);
10341     #endif
10342     fast;
10343
10344     int t = 1;
10345
10346     t = readIntLn(1,1e5);
10347
10348     for(int i=1;i<=t;i++)
10349     {
10350         solve();
10351     }
10352
10353     assert(getchar() == -1);
10354     assert(sum_n<=1e5);
10355
10356     cerr<<"SUCCESS\n";
10357     cerr<<"Tests : " << t << '\n';
10358     cerr<<"Sum of lengths : " << sum_n<<'\n';
10359     //cerr<<"Maximum answer : " << max_n <<'\n';
10360     // // cerr<<"Total operations : " << total_ops << '\n';
10361     // cerr<<"Answered yes : " << yess << '\n';
10362     // cerr<<"Answered no : " << nos << '\n';
10363
10364     cerr << "Time : " << 1000 * ((double)clock()) / (double)CLOCKS_PER_SEC << "ms\n";
10365 }
10366
10367 //ARRSORT-EDITOR
10368 #include <bits/stdc++.h>
10369 using namespace std;
10370 #define pb push_back
10371 #define vi vector<int>
10372 #define rep(i, n) for (int i = 0; i < n; i++)
10373 #define endl '\n'
10374 #define fastio \
10375     ios_base::sync_with_stdio(0); \
10376     cin.tie(0); \
10377     cout.tie(0);

```



```

10378 void tosolve()
10379 {
10380     int n;
10381     cin>>n;
10382     int p[n+1];
10383     for(int i=1;i<=n;i++) cin>>p[i];
10384     int ans=0; //this will store gcd value
10385     for(int i=1;i<=n;i++)
10386     {
10387         ans=__gcd(ans,abs(p[i]-i));
10388     }
10389     cout<<ans<<endl;
10390 }
10391
10392 int32_t main()
10393 {
10394     fastio;
10395     int t;
10396     cin>>t;
10397     while(t--)
10398     {
10399         tosolve();
10400     }
10401     return 0;
10402 }
10403
10404 //PRIME_XOR
10405 #include <bits/stdc++.h>
10406 using namespace std;
10407 #define pb push_back
10408 #define vi vector<int>
10409 #define rep(i, n) for (int i = 0; i < n; i++)
10410 #define endl '\n'
10411 #define fastio \
10412     ios_base::sync_with_stdio(0); \
10413     cin.tie(0); \
10414     cout.tie(0);
10415 void tosolve()
10416 {
10417     vi a;
10418     int b[3];
10419     cin >> b[0] >> b[1];
10420     b[2]=b[0]^b[1];
10421     a.pb(2);
10422     if (b[0] % 2 == 0)
10423     {
10424         a.pb(2^b[1]);
10425         a.pb(2^b[2]);
10426     }
10427     else if (b[1] % 2 == 0)
10428     {
10429         a.pb(2^b[0]);
10430         a.pb(2^b[2]);
10431     }
10432     else if (b[2] % 2 == 0)
10433     {
10434         a.pb(2^b[0]);
10435         a.pb(2^b[1]);
10436     }
10437     sort(a.begin(), a.end());
10438     rep(i,3)
10439     {
10440         if(i==2) cout<<a[i]<<endl;
10441         else cout<<a[i]<<" ";
10442     }
10443     return;
10444 }
10445
10446 int32_t main()

```

```

10447 {
10448     fastio;
10449     int t;
10450     cin>>t;
10451     while(t--)
10452     {
10453         tosolve();
10454     }
10455     return 0;
10456 }
10457
10458 //VOWANX
10459 #include <bits/stdc++.h>
10460 using namespace std;
10461
10462 int main() {
10463     ios::sync_with_stdio(false);
10464     cin.tie(0); cout.tie(0);
10465     int t;
10466     cin >> t;
10467     while(t--) {
10468         int n;
10469         cin >> n;
10470         vector <char> s(n);
10471         for(int i = 0; i < n; i++) {
10472             cin >> s[i];
10473         }
10474         vector <char> t(n);
10475         int l = 0;
10476         int r = n - 1;
10477         bool start = false;
10478         for(int i = n - 1; i >= 0; i--) {
10479             if(start) {
10480                 t[l++] = s[i];
10481             } else {
10482                 t[r--] = s[i];
10483             }
10484             if(s[i] == 'a' || s[i] == 'e' || s[i] == 'i' || s[i] == 'o' || s[i] == 'u') {
10485                 start = !start;
10486             }
10487         }
10488         for(int i = 0; i < n; i++) {
10489             cout << t[i];
10490         }
10491         cout << '\n';
10492     }
10493
10494     return 0;
10495 }
10496
10497 //EQSS
10498 // author: Shivansh Agarwal
10499 #include <bits/stdc++.h>
10500 using namespace std;
10501 #define fastio ios_base::sync_with_stdio(0), cin.tie(0), cout.tie(0)
10502 #define int long long
10503 const int mod = 998244353;
10504
10505 vector<int> T; // Segment Tree
10506 int n;
10507
10508 // 0-indexed
10509 void modify(int p, int value)
10510 {
10511     for (T[p += n] = value; p > 1; p >>= 1)
10512         T[p >> 1] = (T[p] + T[p ^ 1]) % mod;
10513 }
10514
10515 int query(int l, int r) //query [l, r)

```

```

10516 {
10517     int res = 0;
10518     for (l += n, r += n; l < r; l >>= 1, r >>= 1)
10519     {
10520         if (l & 1)
10521             res = (res + T[l++]) % mod;
10522         if (r & 1)
10523             res = (res + T[--r]) % mod;
10524     }
10525     return res;
10526 }
10527
10528 int32_t main()
10529 {
10530     fastio;
10531     int tt;
10532     cin >> tt;
10533     while (tt--)
10534     {
10535         cin >> n;
10536         n *= 2;
10537
10538         // Segment Tree T for range sum queries and point updates
10539         T.clear();
10540         T.resize(2 * n, 0);
10541
10542         //Initializing f and dp vectors which stores the first occurrence of an element
10543         //and the value of dp respectively
10544         vector<int> f(n / 2, -1), dp(n, 0);
10545
10546         for (int i = 0; i < n; i++)
10547         {
10548             int x;
10549             cin >> x;
10550             x--;
10551
10552             //Case 1: f[x] = -1, implies i is the first occurrence
10553             if (f[x] == -1)
10554             {
10555                 f[x] = i;
10556                 if(i > 0)
10557                     dp[i] = dp[i - 1];
10558             }
10559
10560             // Case 2: i is the second occurrence, f[x] is the first
10561             else
10562             {
10563                 //Calculating ti
10564                 int ti = (dp[f[x]] + query(0, f[x]) + 1) % mod;
10565
10566                 //Calculating dp value from ti
10567                 dp[i] = (dp[i - 1] + 2 * ti) % mod;
10568
10569                 //Updating T
10570                 modify(f[x], ti);
10571                 modify(i, mod - ti);
10572             }
10573         }
10574         cout << dp[n - 1] << "\n";
10575     }
10576
10577 //SRTARR
10578 // author: Shivansh Agarwal
10579 #include <bits/stdc++.h>
10580 using namespace std;
10581 #define int long long
10582 #define fastio ios_base::sync_with_stdio(0),cin.tie(0),cout.tie(0)
10583 const int mod = 1000000007;

```

```

10584 const double EPS = 1e-7;
10585
10586
10587
10588 int32_t main()
10589 {
10590     fastio;
10591     auto begin = std::chrono::high_resolution_clock::now();
10592     int tt;
10593     cin >> tt;
10594     while(tt--)
10595     {
10596         int n;
10597         cin >> n;
10598         string s;
10599         cin >> s;
10600         int ans = 0;
10601         for (int i = 0; i < n - 1; i++) if(s[i] == '1' && s[i+1] == '0')
10602             ans++;
10603         cout << ans << "\n";
10604     }
10605     auto end = std::chrono::high_resolution_clock::now();
10606     cerr << setprecision(4) << fixed;
10607     cerr << "Execution time: " <<
        std::chrono::duration_cast<std::chrono::duration<double>>(end - begin).count() << "
        seconds" << endl;
10608 }
10609
10610 //ARMTRN
10611
10612 // Utkarsh Darolia
10613 #include <bits/stdc++.h>
10614 using namespace std;
10615
10616 int main()
10617 {
10618     ios::sync_with_stdio(0);
10619     cin.tie(0);
10620     cout.tie(0);
10621     int t;
10622     cin >> t;
10623     for (int i = 1; i <= t; ++i)
10624     {
10625         int n;
10626         cin >> n;
10627
10628         int TotSum = 0, Sum = 0;
10629         int A[n];
10630
10631         for (int j = 0; j < n; ++j)
10632         {
10633             cin >> A[j];
10634             TotSum += A[j];
10635         }
10636
10637         sort (A, A+n, greater<int>()); //greater<int>() is used for sorting in
        descending order
10638
10639         long long finalAnswer = 0;
10640
10641         for (int r = 0; r < n; ++r)
10642         {
10643             finalAnswer = max (finalAnswer, (long long) (Sum) * ((1000 * (n-r)) - (TotSum-Sum)));
10644             Sum += A[r];
10645         }
10646
10647         cout << finalAnswer << endl;
10648     }
10649     return 0;

```

```

10650 }
10651
10652 //MXDV
10653
10654 // author: Shivansh Agarwal
10655 #include <bits/stdc++.h>
10656 using namespace std;
10657 #define fastio ios_base::sync_with_stdio(0), cin.tie(0), cout.tie(0)
10658
10659 vector<int> st; // Segment Tree
10660 int n;
10661
10662 // 0-based indexing
10663 void MakeST()
10664 {
10665     for (int i = n - 1; i >= 1; i--)
10666         st[i] = __gcd(st[i << 1], st[i << 1 | 1]);
10667 }
10668
10669 int query(int l, int r) // range [l, r)
10670 {
10671     int res = 0;
10672     for (l += n, r += n; l < r; l >>= 1, r >>= 1)
10673     {
10674         if (l & 1)
10675             res = __gcd(res, st[l++]);
10676         if (r & 1)
10677             res = __gcd(res, st[--r]);
10678     }
10679     return res;
10680 }
10681
10682 int32_t main()
10683 {
10684     fastio;
10685     int tt;
10686     cin >> tt;
10687     while (tt--)
10688     {
10689         int x;
10690         cin >> n >> x;
10691         vector<int> v(n);
10692
10693         // Segment Tree for calculating range GCD
10694         st.resize(2 * n);
10695
10696         for (int i = 0; i < n; i++)
10697             cin >> v[i], st[i + n] = v[i];
10698
10699         // building the Segment Tree
10700         MakeST();
10701
10702         // Set for finding the max and second max element of the current subarray
10703         set<pair<int, int>, greater<pair<int, int>>> curr_max;
10704
10705         //Initialising pointers [l, r] for Two-pointers
10706         int l = 0, r = 1, ans = n + 1;
10707
10708         curr_max.insert({v[0], 0});
10709         curr_max.insert({v[1], 1});
10710
10711         // Two Pointers Algorithm
10712         while (r < n && r != l)
10713         {
10714             auto it = curr_max.begin();
10715
10716             // Checking for the largest element
10717             int rest_gcd = __gcd(query(l, (*it).second), query((*it).second + 1, r + 1));
10718             //Here, rest_gcd stores the gcd of rest of elements of the current subarray

```

```

10719         if ((*it).first - rest_gcd >= x)
10720         {
10721             // If condition is satisfied we will update our set and increment left
10722             // pointer
10723             curr_max.erase({v[l], l});
10724             ans = min(ans, r - l + 1), l++;
10725             continue;
10726         }
10727         //Checking for the second largest element
10728         it++;
10729         rest_gcd = __gcd(query(l, (*it).second), query((*it).second + 1, r + 1));
10730         if ((*it).first - rest_gcd >= x)
10731         {
10732             curr_max.erase({v[l], l});
10733             ans = min(ans, r - l + 1), l++;
10734             continue;
10735         }
10736         // If this subarray does not satisfy the condition then we update our set
10737         // and increment right pointer
10738         r++;
10739         if (r < n)
10740             curr_max.insert({v[r], r});
10741     }
10742     // If the value of ans is still n + 1 that means, no subarray satisfied the
10743     // conditions
10744     if (ans == n + 1)
10745         cout << "-1\n";
10746     else
10747         cout << ans << "\n";
10748 }
10749 }
10750
10751 //CRINGEQUERY
10752
10753 #include <bits/stdc++.h>
10754
10755 #include <ext/pb_ds/assoc_container.hpp>
10756 #include <ext/pb_ds/tree_policy.hpp>
10757
10758 using namespace __gnu_pbds;
10759 using namespace std;
10760
10761 template<typename T> using ordered_set = tree <T, null_type, less<T>, rb_tree_tag,
10762 tree_order_statistics_node_update>;
10763
10764 #define _USE_MATH_DEFINES_
10765 #define ll long long
10766 #define ld long double
10767 #define pb push_back
10768 #define mp make_pair
10769 #define sz(x) (int)(x.size())
10770 #define F first
10771 #define S second
10772 #define lb lower_bound
10773 #define ub upper_bound
10774 #define debug(x) cerr << #x << " = " << x << endl
10775 #define SpeedForce ios_base::sync_with_stdio(0), cin.tie(0), cout.tie(0)
10776 #define mt19937 gen<chrono::steady_clock::now().time_since_epoch().count()>;
10777
10778 int rnd (int l, int r) {
10779     return uniform_int_distribution<int> (l, r)(gen);
10780 }
10781 const int N = 1e6+5;
10782 const int mod = 998244353;
10783 int n, q;
10784 vector <int > g[N];

```

```

10784     int used[N];
10785     int comp_size;
10786
10787     int mult(int a, int b) {
10788         return a * (ll) b % mod;
10789     }
10790
10791     int bpow(int a, int b) {
10792         int res = 1;
10793         while(b > 0) {
10794             if(b & 1) res = mult(res, a);
10795             a = mult(a, a);
10796             b >>= 1;
10797         }
10798
10799         return res;
10800     }
10801
10802     void dfs(int v) {
10803         if(used[v])
10804             return;
10805         used[v] = 1;
10806         ++comp_size;
10807         // visiting new node
10808
10809         for (auto to : g[v]) {
10810             dfs(to);
10811         }
10812     }
10813
10814     void solve() {
10815         cin >> n >> q;
10816         for (int i = 1; i <= q; ++i) {
10817             int l, r;
10818             cin >> l >> r;
10819             g[--l].pb(r);
10820             g[r].pb(l);
10821         }
10822         int ans = 1;
10823         for (int i = 0; i <= n; ++i) {
10824             if(used[i])
10825                 continue;
10826             comp_size = 0;
10827             dfs(i);
10828             ans = mult(ans, bpow(2, comp_size - 1));
10829         }
10830         cout << ans << '\n';
10831     }
10832
10833
10834     int32_t main () {
10835         SpeedForce;
10836
10837         int TestCases = 1;
10838         //cin >> TestCases;
10839
10840         for (int TestCase = 1; TestCase <= TestCases; ++TestCase) {
10841             //cout << "Case #" << TestCase << ": ";
10842
10843             solve();
10844         }
10845
10846         return 0;
10847     }
10848
10849     // B...a
10850
10851     //CRINGEQUERY-EDITOR
10852     //Utkarsh.25dec

```

```

10853 #include <bits/stdc++.h>
10854 #define ll long long int
10855 #define pb push_back
10856 #define mp make_pair
10857 #define mod 998244353
10858 #define vl vector <ll>
10859 #define all(c) (c).begin(), (c).end()
10860 using namespace std;
10861 ll power(ll a, ll b) {ll res=1; a%=mod; assert(b>=0);
10862   for(; b>=1;){if(b&1) res=res*a%mod; a=a*a%mod; }return res;}
10863 ll modInverse(ll a){return power(a, mod-2);}
10864 const int N=1000023;
10865 bool vis[N];
10866 vector <int> adj[N];
10867 int cnt=0;
10868 void dfs(int curr)
10869 {
10870     vis[curr]=1;
10871     cnt++;
10872     for(auto it:adj[curr])
10873     {
10874         if(vis[it])
10875             continue;
10876         dfs(it);
10877     }
10878 }
10879 void solve()
10880 {
10881     int n, q;
10882     cin>>n>>q;
10883     ll ans=1;
10884     while(q-->0)
10885     {
10886         int l, r;
10887         cin>>l>>r;
10888         adj[l-1].pb(r);
10889         adj[r].pb(l-1);
10890     }
10891     for(int i=0; i<=n; i++)
10892     {
10893         if(vis[i])
10894             continue;
10895         cnt=0;
10896         dfs(i);
10897         ans*=power(2, cnt-1);
10898         ans%=mod;
10899     }
10900     cout<<ans<<"\n";
10901 }
10902 int main()
10903 {
10904     #ifndef ONLINE_JUDGE
10905     freopen("input.txt", "r", stdin);
10906     freopen("output.txt", "w", stdout);
10907     #endif
10908     ios_base::sync_with_stdio(false);
10909     cin.tie(NULL), cout.tie(NULL);
10910     int T=1;
10911     //cin>>T;
10912     while(T-->0)
10913     {
10914         solve();
10915         cerr << "Time : " << 1000 * ((double)clock()) / (double)CLOCKS_PER_SEC << "ms\n";
10916     }
10917 }
10918 //ARRAYRET
10919 #include <bits/stdc++.h>
10920 #define ll long long int
10921 #define pb push_back
10922 #define mp make_pair

```



```

10921 #define mod 1000000007
10922 #define vl vector <ll>
10923 #define all(c) (c).begin(),(c).end()
10924 using namespace std;
10925 ll power(ll a,ll b) {ll res=1;a%=mod; assert(b>=0);
for(;b;b>>=1){if(b&1)res=res*a%mod;a=a*a%mod;}return res;}
10926 ll modInverse(ll a){return power(a,mod-2);}
10927 const int N=500023;
10928 bool vis[N];
10929 vector <int> adj[N];
10930 long long readInt(long long l,long long r,char endd){
10931     long long x=0;
10932     int cnt=0;
10933     int fi=-1;
10934     bool is_neg=false;
10935     while(true){
10936         char g=getchar();
10937         if(g=='-'){
10938             assert(fi==-1);
10939             is_neg=true;
10940             continue;
10941         }
10942         if('0'<=g && g<='9'){
10943             x*=10;
10944             x+=g-'0';
10945             if(cnt==0){
10946                 fi=g-'0';
10947             }
10948             cnt++;
10949             assert(fi!=0 || cnt==1);
10950             assert(fi!=0 || is_neg==false);
10951
10952             assert(!(cnt>19 || ( cnt==19 && fi>1) ));
10953         } else if(g==endd){
10954             if(is_neg){
10955                 x= -x;
10956             }
10957
10958             if(!(l <= x && x <= r))
10959             {
10960                 cerr << l << ' ' << r << ' ' << x << '\n';
10961                 assert(1 == 0);
10962             }
10963
10964             return x;
10965         } else {
10966             assert(false);
10967         }
10968     }
10969 }
10970 string readString(int l,int r,char endd){
10971     string ret="";
10972     int cnt=0;
10973     while(true){
10974         char g=getchar();
10975         assert(g!=-1);
10976         if(g==endd){
10977             break;
10978         }
10979         cnt++;
10980         ret+=g;
10981     }
10982     assert(l<=cnt && cnt<=r);
10983     return ret;
10984 }
10985 long long readIntSp(long long l,long long r){
10986     return readInt(l,r,' ');
10987 }
10988 long long readIntLn(long long l,long long r){

```

```

10989     return readInt(l,r,'\n');
10990 }
10991 string readStringLn(int l,int r){
10992     return readString(l,r,'\n');
10993 }
10994 string readStringSp(int l,int r){
10995     return readString(l,r,' ');
10996 }
10997 int sumN=0;
10998 void solve()
10999 {
11000     int n=readInt(1,100000,'\n');
11001     sumN+=n;
11002     assert(sumN<=100000);
11003     ll B[n+1]={0};
11004     for(int i=1;i<=n;i++)
11005     {
11006         if(i==n)
11007             B[i]=readInt(1,(ll)200000 * 1000000,'\n');
11008         else
11009             B[i]=readInt(1,(ll)200000 * 1000000,' ');
11010     }
11011     ll sum=0;
11012     for(int i=1;i<=n;i++)
11013         sum+=B[i];
11014     assert((sum%(n+1))==0);
11015     sum/=(n+1);
11016     ll A[n+1]={0};
11017     for(int i=1;i<=n;i++)
11018     {
11019         A[i]=B[i]-sum;
11020         assert(A[i]>=1 && A[i]<=100000);
11021         cout<<A[i]<<' ';
11022     }
11023     cout<<'\n';
11024 }
11025 int main()
11026 {
11027     #ifndef ONLINE_JUDGE
11028     freopen("input.txt", "r", stdin);
11029     freopen("output.txt", "w", stdout);
11030     #endif
11031     ios_base::sync_with_stdio(false);
11032     cin.tie(NULL),cout.tie(NULL);
11033     int T=readInt(1,1000,'\n');
11034     while(T--){
11035         solve();
11036         assert(getchar()==-1);
11037         cerr << "Time : " << 1000 * ((double)clock()) / (double)CLOCKS_PER_SEC << "ms\n";
11038     }
11039
11040 //ARRAYRET-EDITOR
11041 using namespace std;
11042
11043 int main() {
11044     int T;
11045     cin >> T;
11046     while(T--){
11047         int n;
11048         cin >> n;
11049         vector<long long>b(n);
11050         long long big_sum=0;
11051         for(int i=0;i<n;i++){
11052             cin >> b[i];
11053             big_sum+=b[i];
11054         }
11055         long long small_sum=big_sum/(n+1);
11056         for(int i=0;i<n;i++){
11057             cout << b[i]-small_sum << " ";

```

```

11058     }
11059     cout << endl;
11060 }
11061 return 0;
11062 }
11063
11064 //GN THEORY
11065 #include <ext/pb_ds/assoc_container.hpp>
11066 #include <ext/pb_ds/tree_policy.hpp>
11067
11068 using namespace __gnu_pbds;
11069 using namespace std;
11070 using ll = long long;
11071 using ld = long double;
11072
11073 #define fast ios_base::sync_with_stdio(0); cin.tie(0); cout.tie(0);
11074 #define ordered_set tree<int, null_type, less<int>,
rb_tree_tag, tree_order_statistics_node_update>
11075
11076 #define pb push_back
11077 #define mp make_pair
11078 #define F first
11079 #define S second
11080 #define pii pair<int , int>
11081 #define int long long int
11082 #define endl "\n"
11083
11084
11085 #define ALL(v) v.begin(),v.end()
11086 #define ALLR(v) v.rbegin(),v.rend()
11087 // #define sz(v) (int) v.size()
11088 #define PI 3.14159265358979323
11089 #define inf LLONG_MAX
11090 #define ones(x) __builtin_popcount(x)
11091 #define mod 1000000007
11092 #define MOD 998244353
11093
11094 ll mod_pow(ll a,ll b,ll m)
11095 {
11096     ll res = 1;
11097     while(b)
11098     {
11099         if(b&1)
11100         {
11101             res=(res*a) % m;
11102         }
11103         a=(a*a) % m;
11104         b>>=1;
11105     }
11106     return res;
11107 }
11108
11109 ll mod_inverse(int a , int m)
11110 {
11111     return mod_pow(a , m - 2 , m);
11112 }
11113
11114 const int N = 1e5 + 5;
11115
11116 int lpf[N];
11117
11118 void pre() {
11119     for(int i = 2; i < N; ++i) {
11120         if(lpf[i] == 0) {
11121             for(int j = i; j < N; j += i) {
11122                 lpf[j] = i;
11123             }
11124         }
11125     }

```

```

11126     }
11127
11128 void solve()
11129 {
11130     int n , q;
11131
11132     cin >> n >> q;
11133
11134     for(int i = 0; i < q; ++i) {
11135         int u , v;
11136
11137         cin >> u >> v;
11138
11139         int dummy_u = u;
11140         int dummy_v = v;
11141
11142         map<int , int> frq;
11143
11144         while(dummy_u > 1) {
11145             int y = lpf[dummy_u];
11146             while(dummy_u % y == 0) {
11147                 dummy_u /= y;
11148                 frq[y] += 1;
11149             }
11150         }
11151
11152         while(dummy_v > 1) {
11153             int y = lpf[dummy_v];
11154             while(dummy_v % y == 0) {
11155                 dummy_v /= y;
11156                 frq[y] -= 1;
11157             }
11158         }
11159
11160         int res = 0;
11161
11162         for(auto r : frq) {
11163             res += abs(r.S) * r.F;
11164         }
11165
11166         cout << res << endl;
11167     }
11168 }
11169
11170 signed main() {
11171     fast;
11172
11173     int t = 1;
11174
11175     pre();
11176
11177     cin >> t;
11178
11179     while(t--) {
11180         solve();
11181     }
11182
11183     return 0;
11184 }
11185
11186 //GN_THEORY-EDITOR
11187 #define MAXN 100001
11188 using namespace std;
11189 int spf[MAXN];
11190 void sieve() {
11191     spf[1]=1;
11192     for(int i=2;i<MAXN;i++) spf[i]=i;
11193     for(int i=4;i<MAXN;i+=2) spf[i]=2;
11194     for(int i=3;i*i<MAXN;i++){

```

```

11195         if(spfa[i]==i){
11196             for(int j=i*i;j<MAXN;j+=i)
11197                 if(spfa[j]==j)
11198                     spfa[j]=i;
11199         }
11200     }
11201 }
11202 vector<int> getFactorization(int x){
11203     vector<int>ret;
11204     while(x!=1){
11205         ret.push_back(spfa[x]);
11206         x=x/spfa[x];
11207     }
11208     return ret;
11209 }
11210 int main() {
11211     sieve();
11212     int T;
11213     cin >> T;
11214     while(T--){
11215         int N,Q;
11216         cin >> N >> Q;
11217         for(int i=0;i<Q;i++){
11218             long long ans=0;
11219             int U,V;
11220             cin >> U >> V;
11221             vector<int>res1=getFactorization(U);
11222             map<int,int>ff;
11223             for(auto x:res1)ff[x]++;
11224             vector<int>res2=getFactorization(V);
11225             for(auto x:res2)ff[x]--;
11226             for(auto x:ff){
11227                 ans+=abs(x.second)*x.first;
11228             }
11229             cout << ans << endl;
11230         }
11231     }
11232     return 0;
11233 }
11234
11235 //XORMUL
11236 using namespace std;
11237
11238 int main()
11239 {
11240     int T; cin >> T;
11241     while(T--){
11242     {
11243         int n, a, b; cin >> n >> a >> b;
11244         int ans = 0;
11245         bool fst = 1;
11246         for(int i = n - 1; i >= 0; i--){
11247             {
11248                 bool A = a >> i & 1;
11249                 bool B = b >> i & 1;
11250                 if(!A and !B)
11251                     ans |= 1 << i;
11252                 if(A and !B)
11253                 {
11254                     if(fst)
11255                         fst = false;
11256                     else
11257                         ans |= 1 << i;
11258                 }
11259                 if(!A and B)
11260                 {
11261                     if(fst)
11262                         ans |= 1 << i, fst = false;
11263                     else

```

```

11264         ;
11265     }
11266 }
11267     cout << ans << endl;
11268 }
11269     return 0;
11270 }
11271
11272 //XORMUL-EDITOR
11273 using namespace std;
11274
11275 int main() {
11276     int T;
11277     cin >> T;
11278     while(T--){
11279         int n,a,b;
11280         cin >> n >> a >> b;
11281         int c=0;
11282         bool ok=true;
11283         for(int i=n-1;i>=0;i--){
11284             if(((1<<i)&a)==((1<<i)&b)){
11285                 if(((1<<i)&a)==0)c+=(1<<i);
11286             }
11287             else if(ok){
11288                 if(((1<<i)&b))c+=(1<<i);
11289                 ok=false;
11290             }
11291             else{
11292                 if(((1<<i)&a))c+=(1<<i);
11293             }
11294         }
11295         cout << c << endl;
11296     }
11297     return 0;
11298 }
11299
11300 //EVEQODD
11301 using namespace std;
11302
11303 int main() {
11304     int tt;
11305     cin >> tt;
11306     while (tt--){
11307         int n;
11308         cin >> n;
11309         vector<int> a(2 * n);
11310         for (int i = 0; i < 2 * n; i++) {
11311             cin >> a[i];
11312         }
11313         vector<int> p2(2 * n);
11314         for (int i = 0; i < 2 * n; i++) {
11315             while (a[i] % 2 == 0) {
11316                 a[i] /= 2;
11317                 p2[i]++;
11318             }
11319         }
11320         sort(p2.begin(), p2.end());
11321         int odds = (int) count(p2.begin(), p2.end(), 0);
11322         int ans = 0;
11323         if (odds <= n) {
11324             for (int i = 0; i < n; i++) {
11325                 ans += p2[i];
11326             }
11327         } else {
11328             ans = odds - n;
11329         }
11330         cout << ans << endl;
11331     }
11332     return 0;

```

```

11333     }
11334
11335     //EVEQODD-EDITOR
11336     using namespace std;
11337
11338     int main() {
11339         int T;
11340         cin >> T;
11341         while(T--){
11342             int n;
11343             cin >> n;
11344             n*=2;
11345             vector<int>a(n);
11346             int e=0,o=0;
11347             vector<int>make_o;
11348             for(int i=0;i<n;i++){
11349                 cin >> a[i];
11350                 if(a[i]%2)o++;
11351                 else{
11352                     e++;
11353                     int x=0;
11354                     while(a[i]%2==0){
11355                         x++;
11356                         a[i]/=2;
11357                     }
11358                     make_o.push_back(x);
11359                 }
11360             }
11361             sort(make_o.begin(),make_o.end());
11362             if(o>=e)cout << (o-e)/2 << endl;
11363             else{
11364                 int ans=0;
11365                 for(int i=0;i<(e-o)/2;i++)ans+=make_o[i];
11366                 cout << ans << endl;
11367             }
11368         }
11369         return 0;
11370     }
11371
11372     //NDANDANDOR
11373     #include <bits/stdc++.h>
11374     using namespace std;
11375
11376     #define nl "\n"
11377     #define nf endl
11378     #define ll long long
11379     #define pb push_back
11380     #define _ << ' ' <<
11381
11382     #define INF (ll)1e18
11383     #define mod 998244353
11384     #define maxn 2097162
11385
11386     ll i, il, j, k, k1, t, n, m, res, flag[10], a, b;
11387     ll fc[maxn], nv[maxn], c[2], d[2], bb;
11388
11389     ll fxp(ll b, ll e) {
11390         ll r = 1, k = b;
11391         while (e != 0) {
11392             if (e % 2) r = (r * k) % mod;
11393             k = (k * k) % mod; e /= 2;
11394         }
11395         return r;
11396     }
11397
11398     ll inv(ll x) {
11399         return fxp(x, mod - 2);
11400     }
11401

```

```

11402 ll bnm(ll a, ll b) {
11403     if (a < b || b < 0) return 0;
11404     ll r = (fc[a] * nv[b]) % mod;
11405     r = (r * nv[a - b]) % mod;
11406     return r;
11407 }
11408
11409
11410 int main() {
11411     ios::sync_with_stdio(0);
11412     cin.tie(0);
11413
11414     fc[0] = 1; nv[0] = 1;
11415     for (i = 1; i < maxn; i++) {
11416         fc[i] = (i * fc[i - 1]) % mod; nv[i] = inv(fc[i]);
11417     }
11418
11419     cin >> t;
11420     while (t--) {
11421         cin >> n >> m >> k;
11422         c[0] = 0; c[1] = 0; d[0] = 1; d[1] = 1; res = 0;
11423         for (i = 0; i <= 19; i++) d[(k >> i) & 1] *= 2;
11424         for (i = 19; i >= -1; i--) {
11425             if (i != -1) {
11426                 c[(k >> i) & 1] *= 2; d[(k >> i) & 1] /= 2;
11427                 if ((m >> i) & 1) == 0) continue;
11428             }
11429             res = (res + (bnm((c[0] + 1) * d[0] + n - 1, n) - bnm(c[0] * d[0] + n - 1,
11430                 (bnm((c[1] + 1) * d[1] + n - 1, n) - bnm(c[1] * d[1] + n - 1, n) +
11431                     mod)) % mod;
11432             // cout << "i, c[0], c[1], d[0], d[1], res =" _ i _ c[0] _ c[1] _ d[0] _
11433             d[1] _ res << nl;
11434             if (i != -1) c[(k >> i) & 1]++;
11435         }
11436         cout << res << nl;
11437     }
11438     return 0;
11439 }
11440
11441 //NDANDANDOR-EDITOR
11442
11443 //Utkarsh.25dec
11444 #include <bits/stdc++.h>
11445 #define ll long long int
11446 #define pb push_back
11447 #define mp make_pair
11448 #define mod 998244353
11449 #define vl vector <ll>
11450 #define all(c) (c).begin(), (c).end()
11451 using namespace std;
11452 ll power(ll a, ll b) {ll res=1; a%=mod; assert(b>=0);
11453     for(; b; b>>=1) {if(b&1) res=res*a%mod; a=a*a%mod;} return res;}
11454 ll modInverse(ll a) {return power(a, mod-2);}
11455 const int N=4000023;
11456 bool vis[N];
11457 vector <int> adj[N];
11458 ll fact[N];
11459 ll invfact[N];
11460 ll inv[N];
11461 void factorialsComputation()
11462 {
11463     inv[0]=inv[1]=1;
11464     fact[0]=fact[1]=1;
11465     invfact[0]=invfact[1]=1;
11466     for(int i=2; i<N; i++)

```



```

11467         inv[i]=(inv[mod%i]*(mod-mod/i))%mod;
11468         fact[i]=(fact[i-1]*i)%mod;
11469         invfact[i]=(invfact[i-1]*inv[i])%mod;
11470     }
11471 }
11472 ll ncr(ll n,ll r)
11473 {
11474     ll ans=fact[n]*invfact[r];
11475     ans%=mod;
11476     ans*=invfact[n-r];
11477     ans%=mod;
11478     return ans;
11479 }
11480 void solve()
11481 {
11482     ll n,m,k;
11483     cin>>n>>m>>k;
11484     m++;
11485     int high=0;
11486     for(int i=22;i>=0;i--)
11487     {
11488         if((m&(1<<i))!=0)
11489         {
11490             high=i;
11491             break;
11492         }
11493     }
11494     ll ans=0;
11495     for(int i=high;i>=0;i--)
11496     {
11497         if((m&(1<<i))==0)
11498             continue;
11499         ll x=0,y=0;
11500         for(int j=high;j>=i;j--)
11501         {
11502             if(j==i)
11503             {
11504                 if((k&(1<<j))==0)
11505                     x*=2;
11506                 else
11507                     y*=2;
11508                 continue;
11509             }
11510             if((k&(1<<j))==0)
11511             {
11512                 x*=2;
11513                 if((m&(1<<j))!=0)
11514                     x++;
11515             }
11516             else
11517             {
11518                 y*=2;
11519                 if((m&(1<<j))!=0)
11520                     y++;
11521             }
11522         }
11523         ll p=0,q=0;
11524         for(int j=i-1;j>=0;j--)
11525         {
11526             if((k&(1<<j))==0)
11527             {
11528                 p*=2;
11529                 x*=2;
11530                 p++;
11531             }
11532             else
11533             {
11534                 q*=2;
11535                 y*=2;

```

```

11536         q++;
11537     }
11538 }
11539 ll tmp=(ncr(n+x+p,n)+mod-ncr(n+x-1,n))*(ncr(n+y+q,n)+mod-ncr(n+y-1,n));
11540 tmp%=mod;
11541 ans+=tmp;
11542 ans%=mod;
11543 }
11544 cout<<ans<<'\n';
11545 }
11546 int main()
11547 {
11548     #ifndef ONLINE_JUDGE
11549     freopen("input.txt", "r", stdin);
11550     freopen("output.txt", "w", stdout);
11551     #endif
11552     ios_base::sync_with_stdio(false);
11553     cin.tie(NULL),cout.tie(NULL);
11554     factorialsComputation();
11555     int T=1;
11556     cin>>T;
11557     while(T--)
11558         solve();
11559     cerr << "Time : " << 1000 * ((double)clock()) / (double)CLOCKS_PER_SEC << "ms\n";
11560 }
11561
11562 //START44-EXTERNALSITE
11563 //HOWMANYMAX
11564 #ifdef WTSH
11565     #include <wtsh.h>
11566 #else
11567     #include <bits/stdc++.h>
11568     using namespace std;
11569     #define dbg(...)
11570 #endif
11571
11572 #define int long long
11573 #define endl "\n"
11574 #define sz(w) (int)(w.size())
11575 using pii = pair<int, int>;
11576
11577 const long long INF = 1e18;
11578
11579 const int N = 1e6 + 5;
11580
11581 // ----- Input Checker Start -----
11582
11583 long long readInt(long long l, long long r, char endd)
11584 {
11585     long long x = 0;
11586     int cnt = 0, fi = -1;
11587     bool is_neg = false;
11588     while(true)
11589     {
11590         char g = getchar();
11591         if(g == '-')
11592         {
11593             assert(fi == -1);
11594             is_neg = true;
11595             continue;
11596         }
11597         if('0' <= g && g <= '9')
11598         {
11599             x *= 10;
11600             x += g - '0';
11601             if(cnt == 0)
11602                 fi = g - '0';
11603             cnt++;
11604             assert(fi != 0 || cnt == 1);

```

```

11605         assert(fi != 0 || is_neg == false);
11606         assert(!(cnt > 19 || (cnt == 19 && fi > 1)));
11607     }
11608     else if(g == endd)
11609     {
11610         if(is_neg)
11611             x = -x;
11612         if(!(l <= x && x <= r))
11613         {
11614             cerr << l << ' ' << r << ' ' << x << '\n';
11615             assert(false);
11616         }
11617         return x;
11618     }
11619     else
11620     {
11621         assert(false);
11622     }
11623 }
11624 }
11625
11626 string readString(int l, int r, char endd)
11627 {
11628     string ret = "";
11629     int cnt = 0;
11630     while(true)
11631     {
11632         char g = getchar();
11633         assert(g != -1);
11634         if(g == endd)
11635             break;
11636         cnt++;
11637         ret += g;
11638     }
11639     assert(l <= cnt && cnt <= r);
11640     return ret;
11641 }
11642
11643 long long readIntSp(long long l, long long r) { return readInt(l, r, ' '); }
11644 long long readIntLn(long long l, long long r) { return readInt(l, r, '\n'); }
11645 string readStringLn(int l, int r) { return readString(l, r, '\n'); }
11646 string readStringSp(int l, int r) { return readString(l, r, ' '); }
11647 void readEOF() { assert(getchar() == EOF); }
11648
11649 vector<int> readVectorInt(int n, long long l, long long r)
11650 {
11651     vector<int> a(n);
11652     for(int i = 0; i < n - 1; i++)
11653         a[i] = readIntSp(l, r);
11654     a[n - 1] = readIntLn(l, r);
11655     return a;
11656 }
11657
11658 // ----- Input Checker End -----
11659
11660 int sumN = 0;
11661
11662 void solve()
11663 {
11664     int n = readIntLn(2, 1e5);
11665     sumN += n;
11666     string s = readStringLn(n - 1, n - 1);
11667     assert(*min_element(s.begin(), s.end()) >= '0' and *max_element(s.begin(), s.end())
11668         <= '1');
11669     s = '0' + s + '1';
11670     int ans = 0;
11671     for(int i = 0; i + 1 < sz(s); i++)
11672     {
11673         if(s[i] == '0' and s[i + 1] == '1')

```

```

11673         ans++;
11674     }
11675     cout << ans << endl;
11676 }
11677
11678 int32_t main()
11679 {
11680     ios::sync_with_stdio(0);
11681     cin.tie(0);
11682     int T = readIntLn(1, 1e5);
11683     for(int tc = 1; tc <= T; tc++)
11684     {
11685         // cout << "Case #" << tc << ": ";
11686         solve();
11687     }
11688     assert(sumN <= 1e5);
11689     readEOF();
11690     return 0;
11691 }
11692
11693 //HOWMANYMAX-EDITOR
11694 #include "bits/stdc++.h"
11695 using namespace std;
11696 #define ll long long
11697 #define pb push_back
11698 #define all(_obj) _obj.begin(), _obj.end()
11699 #define F first
11700 #define S second
11701 #define pll pair<ll, ll>
11702 #define vll vector<ll>
11703 ll INF = 1e18;
11704 const int N = 1e5 + 11, mod = 1e9 + 7;
11705 ll max(ll a, ll b) { return ((a > b) ? a : b); }
11706 ll min(ll a, ll b) { return ((a > b) ? b : a); }
11707 mt19937 rng(chrono::steady_clock::now().time_since_epoch().count());
11708 void sol(void)
11709 {
11710     int n, ans = 0;
11711     cin >> n;
11712     string s;
11713     cin >> s;
11714     for (int i = 0; i < n - 1; i++)
11715     {
11716         if (s[i] == '1' && (i == 0 || s[i - 1] == '0'))
11717             ans++;
11718     }
11719     if (s.back() == '0')
11720         ans++;
11721     cout << ans << '\n';
11722     return;
11723 }
11724 int main()
11725 {
11726     ios_base::sync_with_stdio(false);
11727     cin.tie(NULL), cout.tie(NULL);
11728     int test = 1;
11729     cin >> test;
11730     while (test--)
11731         sol();
11732 }
11733
11734 //MINFLIPS
11735 //Utkarsh.25dec
11736 #include <bits/stdc++.h>
11737 #define ll long long int
11738 #define pb push_back
11739 #define mp make_pair
11740 #define mod 1000000007
11741 #define vl vector<ll>

```

```

11742 #define all(c) (c).begin(), (c).end()
11743 using namespace std;
11744 ll power(ll a, ll b) {ll res=1; a%=mod; assert(b>=0);
11745 for(; b; b>>=1) {if(b&1) res=res*a%mod; a=a*a%mod;} return res;}
11746 ll modInverse(ll a) {return power(a, mod-2);}
11747 const int N=500023;
11748 bool vis[N];
11749 vector<int> adj[N];
11750 long long readInt(long long l, long long r, char endd) {
11751     long long x=0;
11752     int cnt=0;
11753     int fi=-1;
11754     bool is_neg=false;
11755     while(true) {
11756         char g=getchar();
11757         if(g=='-') {
11758             assert(fi===-1);
11759             is_neg=true;
11760             continue;
11761         }
11762         if('0'<=g && g<='9') {
11763             x*=10;
11764             x+=g-'0';
11765             if(cnt==0) {
11766                 fi=g-'0';
11767             }
11768             cnt++;
11769             assert(fi!=0 || cnt==1);
11770             assert(fi!=0 || is_neg==false);
11771
11772             assert(!(cnt>19 || ( cnt==19 && fi>1) ));
11773         } else if(g==endd) {
11774             if(is_neg) {
11775                 x= -x;
11776             }
11777
11778             if(!(l <= x && x <= r))
11779             {
11780                 cerr << l << ' ' << r << ' ' << x << '\n';
11781                 assert(1 == 0);
11782             }
11783
11784             return x;
11785         } else {
11786             assert(false);
11787         }
11788     }
11789 string readString(int l, int r, char endd) {
11790     string ret="";
11791     int cnt=0;
11792     while(true) {
11793         char g=getchar();
11794         assert(g!=-1);
11795         if(g==endd) {
11796             break;
11797         }
11798         cnt++;
11799         ret+=g;
11800     }
11801     assert(l<=cnt && cnt<=r);
11802     return ret;
11803 }
11804 long long readIntSp(long long l, long long r) {
11805     return readInt(l, r, ' ');
11806 }
11807 long long readIntLn(long long l, long long r) {
11808     return readInt(l, r, '\n');
11809 }

```

```

11810 string readStringLn(int l,int r){
11811     return readString(l,r,'\n');
11812 }
11813 string readStringSp(int l,int r){
11814     return readString(l,r,' ');
11815 }
11816 void solve()
11817 {
11818     int N=readInt(2,1000,'\n');
11819     int A[N+1]={0};
11820     int sum=0;
11821     for(int i=1;i<=N;i++)
11822     {
11823         if(i==N)
11824             A[i]=readInt(-1,1,'\n');
11825         else
11826             A[i]=readInt(-1,1,' ');
11827         assert(A[i]!=0);
11828         sum+=A[i];
11829     }
11830     if(N%2==1)
11831     {
11832         cout<<-1<<'\n';
11833         return;
11834     }
11835     sum=abs(sum);
11836     cout<<(sum/2)<<'\n';
11837 }
11838 int main()
11839 {
11840     #ifndef ONLINE_JUDGE
11841     freopen("input.txt", "r", stdin);
11842     freopen("output.txt", "w", stdout);
11843     #endif
11844     ios_base::sync_with_stdio(false);
11845     cin.tie(NULL),cout.tie(NULL);
11846     int T=readInt(1,100,'\n');
11847     while(T--)
11848         solve();
11849     assert(getchar()==-1);
11850     cerr << "Time : " << 1000 * ((double)clock()) / (double)CLOCKS_PER_SEC << "ms\n";
11851 }
11852
11853 //MINFLIPS-EDITOR
11854 #include "bits/stdc++.h"
11855 using namespace std;
11856 #define ll long long
11857 #define pb push_back
11858 #define all(_obj) _obj.begin(), _obj.end()
11859 #define F first
11860 #define S second
11861 #define pll pair<ll, ll>
11862 #define vll vector<ll>
11863 ll INF = 1e18;
11864 const int N = 1e5 + 11, mod = 1e9 + 7;
11865 ll max(ll a, ll b) { return ((a > b) ? a : b); }
11866 ll min(ll a, ll b) { return ((a > b) ? b : a); }
11867 mt19937 rng(chrono::steady_clock::now().time_since_epoch().count());
11868 void sol(void)
11869 {
11870     int n;
11871     cin >> n;
11872     int cnt1 = 0;
11873     vll v(n);
11874     for (int i = 0; i < n; i++)
11875     {
11876         cin >> v[i];
11877         cnt1 += (v[i] == 1);
11878     }

```

```

11879     if (n & 1)
11880         cout << -1 << '\n';
11881     else if (cnt1 >= n / 2)
11882         cout << cnt1 - n / 2 << '\n';
11883     else
11884         cout << n / 2 - cnt1 << '\n';
11885
11886     return;
11887 }
11888 int main()
11889 {
11890     ios_base::sync_with_stdio(false);
11891     cin.tie(NULL), cout.tie(NULL);
11892     int test = 1;
11893     cin >> test;
11894     while (test--)
11895         sol();
11896 }
11897
11898 //MODULO3
11899 #include <bits/stdc++.h>
11900 #include <ext/pb_ds/tree_policy.hpp>
11901 #include <ext/pb_ds/assoc_container.hpp>
11902 using namespace __gnu_pbds;
11903 using namespace std;
11904 #define ll long long
11905 const ll INF_MUL=1e13;
11906 const ll INF_ADD=1e18;
11907 #define pb push_back
11908 #define mp make_pair
11909 #define nline "\n"
11910 #define f first
11911 #define s second
11912 #define pll pair<ll,ll>
11913 #define vl vector<ll>
11914 #define vvl vector<vector<ll>>
11915 #define vvvl vector<vector<vector<ll>>>
11916 #define all(v) v.begin(),v.end()
11917 #ifndef ONLINE_JUDGE
11918 #define debug(x) cerr<<#x<<" "; _print(x); cerr<<nline;
11919 #else
11920 #define debug(x);
11921 #endif
11922 void _print(ll x){cerr<<x;}
11923 void _print(string x){cerr<<x;}
11924 mt19937 rng(chrono::steady_clock::now().time_since_epoch().count());
11925 template<class T,class V> void _print(pair<T,V> p) {cerr<<"{";
11926     _print(p.first);cerr<<","; _print(p.second);cerr<<"}";}
11927 template<class T>void _print(vector<T> v) {cerr<<" [ "; for (T i:v){_print(i);cerr<<"
11928     ";}cerr<<"}";}
11929 template<class T>void _print(set<T> v) {cerr<<" [ "; for (T i:v){_print(i); cerr<<"
11930     ";}cerr<<"}";}
11931 template<class T>void _print(multiset<T> v) {cerr<<" [ "; for (T
11932     i:v){_print(i);cerr<<" ";}cerr<<"}";}
11933 template<class T,class V>void _print(map<T, V> v) {cerr<<" [ "; for(auto i:v)
11934     {_print(i);cerr<<" ";} cerr<<"}";}
11935 template<class T> using oset=tree<T, null_type, less<T>, rb_tree_tag,
11936     tree_order_statistics_node_update>;
11937 template<class T> using muloset=tree<T, null_type, less_equal<T>, rb_tree_tag,
11938     tree_order_statistics_node_update>;
11939 //-----
11940 -----
11941
11942 const ll MOD=1e9+7;
11943 const ll MAX=500500;
11944 vector<ll> value(MAX);
11945 vector<vector<ll>> adj;
11946 vector<multiset<ll>> track(MAX);
11947 void dfs(ll cur,ll par){

```

```

11939     for(auto chld:adj[cur]){
11940         if(chld!=par){
11941             dfs(chld,cur);
11942             if(track[chld].size()>track[cur].size()){
11943                 swap(track[chld],track[cur]);
11944             }
11945             for(auto i:track[chld]){
11946                 track[cur].insert(i);
11947             }
11948         }
11949     }
11950     if(track[cur].empty()){
11951         track[cur].insert(value[cur]);
11952     }
11953     else{
11954         auto fs=track[cur].begin();
11955         ll val=*fs;
11956         if(val<value[cur]){
11957             track[cur].erase(fs);
11958             track[cur].insert(value[cur]);
11959         }
11960     }
11961 }
11962 void solve(){
11963     ll a,b; cin>>a>>b;
11964     a=a%3,b=b%3;
11965     if(min(a,b)==0){
11966         cout<<"0\n";
11967     }
11968     else if(a==b){
11969         cout<<"1\n";
11970     }
11971     else{
11972         cout<<"2\n";
11973     }
11974     return;
11975 }
11976 int main()
11977 {
11978     ios_base::sync_with_stdio(false);
11979     cin.tie(NULL);
11980     #ifndef ONLINE_JUDGE
11981     freopen("input.txt", "r", stdin);
11982     freopen("output.txt", "w", stdout);
11983     freopen("error.txt", "w", stderr);
11984     #endif
11985     ll test_cases=1;
11986     cin>>test_cases;
11987     while(test_cases--){
11988         solve();
11989     }
11990     cout<<fixed<<setprecision(15);
11991     cerr<<"Time:"<<1000*((double)clock())/((double)CLOCKS_PER_SEC)<<"ms\n";
11992 }
11993
11994 //MODULO3-EDITOR
11995 #include "bits/stdc++.h"
11996 using namespace std;
11997 #define ll long long
11998 #define pb push_back
11999 #define all(_obj) _obj.begin(), _obj.end()
12000 #define F first
12001 #define S second
12002 #define pll pair<ll, ll>
12003 #define vll vector<ll>
12004 ll INF = 1e18;
12005 const int N = 1e5 + 11, mod = 1e9 + 7;
12006 ll max(ll a, ll b) { return ((a > b) ? a : b); }
12007 ll min(ll a, ll b) { return ((a > b) ? b : a); }

```



```

12008 mt19937 rng(chrono::steady_clock::now().time_since_epoch().count());
12009 void sol(void)
12010 {
12011     int a, b;
12012     cin >> a >> b;
12013     if (a % 3 == 0 || b % 3 == 0)
12014         cout << 0 << '\n';
12015     else if (a % 3 == b % 3)
12016         cout << 1 << '\n';
12017     else
12018         cout << 2 << '\n';
12019     return;
12020 }
12021 int main()
12022 {
12023     ios_base::sync_with_stdio(false);
12024     cin.tie(NULL), cout.tie(NULL);
12025     int test = 1;
12026     cin >> test;
12027     while (test--)
12028         sol();
12029 }
12030
12031 //CAARRAY
12032 #include <bits/stdc++.h>
12033 #include <ext/pb_ds/tree_policy.hpp>
12034 #include <ext/pb_ds/assoc_container.hpp>
12035 using namespace __gnu_pbds;
12036 using namespace std;
12037 #define ll long long
12038 const ll INF_MUL=1e13;
12039 const ll INF_ADD=1e18;
12040 #define pb push_back
12041 #define mp make_pair
12042 #define nline "\n"
12043 #define f first
12044 #define s second
12045 #define pll pair<ll,ll>
12046 #define vl vector<ll>
12047 #define vvl vector<vector<ll>>
12048 #define vvvl vector<vector<vector<ll>>>
12049 #define all(v) v.begin(),v.end()
12050 #ifndef ONLINE_JUDGE
12051 #define debug(x) cerr<<#x<<" "; _print(x); cerr<<nline;
12052 #else
12053 #define debug(x);
12054 #endif
12055 void _print(ll x){cerr<<x;}
12056 void _print(string x){cerr<<x;}
12057 mt19937 rng(chrono::steady_clock::now().time_since_epoch().count());
12058 template<class T,class V> void _print(pair<T,V> p) {cerr<< "{";
12059     _print(p.first);cerr<<","; _print(p.second);cerr<<"}";}
12060 template<class T>void _print(vector<T> v) {cerr<< " [ "; for (T i:v){_print(i);cerr<<
12061     ";}cerr<<"]";}
12062 template<class T>void _print(set<T> v) {cerr<< " [ "; for (T i:v){_print(i); cerr<<
12063     ";}cerr<<"]";}
12064 template<class T>void _print(multiset<T> v) {cerr<< " [ "; for (T
12065     i:v){_print(i);cerr<<"";cerr<<"]";}
12066 template<class T,class V>void _print(map<T, V> v) {cerr<< " [ "; for(auto i:v)
12067     {_print(i);cerr<<"";cerr<<"]";}
12068 template<class T> using oset=tree<T, null_type, less<T>, rb_tree_tag,
12069     tree_order_statistics_node_update>;
12070 template<class T> using muloset=tree<T, null_type, less_equal<T>, rb_tree_tag,
12071     tree_order_statistics_node_update>;
12072 //-----
12073 -----
12074 const ll MOD=1e9+7;
12075 const ll MAX=500500;

```

```

12068 void solve() {
12069     ll n; cin>>n;
12070     vector<ll> a(n+5,0);
12071     ll pos=1;
12072     for(ll i=(n%4)+1;i<=n-3;i+=4){
12073         a[pos++]=i*(i+3),a[pos++]=(i+1)*(i+3);
12074         a[pos++]=(i+1)*(i+2),a[pos++]=(i+2)*(i+2);
12075     }
12076     ll ext=1;
12077     for(ll i=2;i<=(n%4);i++){
12078         a[pos++]=i;
12079         ext*=i;
12080     }
12081     if(pos==n){
12082         a[pos]=ext;
12083     }
12084     for(ll i=1;i<=n;i++){
12085         cout<<a[i]<<" ";
12086     }
12087     cout<<endl;
12088     return;
12089 }
12090 int main()
12091 {
12092     ios_base::sync_with_stdio(false);
12093     cin.tie(NULL);
12094     #ifndef ONLINE_JUDGE
12095     freopen("input.txt", "r", stdin);
12096     freopen("output.txt", "w", stdout);
12097     freopen("error.txt", "w", stderr);
12098     #endif
12099     ll test_cases=1;
12100     cin>>test_cases;
12101     while(test_cases--){
12102         solve();
12103     }
12104     cout<<fixed<<setprecision(15);
12105     cerr<<"Time:"<<1000*((double)clock())/(double)CLOCKS_PER_SEC<<"ms\n";
12106 }
12107
12108 //CAARRAY-EDITOR
12109 #include "bits/stdc++.h"
12110 using namespace std;
12111 #define ll long long
12112 #define pb push_back
12113 #define all(_obj) _obj.begin(), _obj.end()
12114 #define F first
12115 #define S second
12116 #define pll pair<ll, ll>
12117 #define vll vector<ll>
12118 ll INF = 1e18;
12119 const int N = 1e5 + 11, mod = 1e9 + 7;
12120 ll max(ll a, ll b) { return ((a > b) ? a : b); }
12121 ll min(ll a, ll b) { return ((a > b) ? b : a); }
12122 mt19937 rng(chrono::steady_clock::now().time_since_epoch().count());
12123 void sol(void)
12124 {
12125     int n;
12126     cin >> n;
12127     if (n == 3)
12128     {
12129         cout << 2 << ' ' << 3 << ' ' << 6 << '\n';
12130         return;
12131     }
12132     for (int i = n; i >= 1; i -= 4)
12133     {
12134         if (i <= 3)
12135             for (int j = 1; j <= i; j++)
12136                 cout << j << ' ';

```

```

12137         else
12138         {
12139             int a = i - 3;
12140             cout << a * (a + 3) << ' ' << a * (a + 2) << ' ' << (a + 1) * (a + 3) << ' ' << (a + 1) * (a + 2) << ' ';
12141         }
12142     }
12143     cout << '\n';
12144     return;
12145 }
12146 int main()
12147 {
12148     ios_base::sync_with_stdio(false);
12149     cin.tie(NULL), cout.tie(NULL);
12150     int test = 1;
12151     cin >> test;
12152     while (test--)
12153         sol();
12154 }
12155
12156 //MAXIMISEBITS
12157 #include <bits/stdc++.h>
12158 #include <ext/pb_ds/tree_policy.hpp>
12159 #include <ext/pb_ds/assoc_container.hpp>
12160 using namespace __gnu_pbds;
12161 using namespace std;
12162 #define ll long long
12163 const ll INF_MUL=1e13;
12164 const ll INF_ADD=1e18;
12165 #define pb push_back
12166 #define mp make_pair
12167 #define nline "\n"
12168 #define f first
12169 #define s second
12170 #define pll pair<ll,ll>
12171 #define vl vector<ll>
12172 #define vvl vector<vector<ll>>
12173 #define vvvl vector<vector<vector<ll>>>
12174 #define all(v) v.begin(),v.end()
12175 #ifndef ONLINE_JUDGE
12176 #define debug(x) cerr<<#x<<" "; _print(x); cerr<<nline;
12177 #else
12178 #define debug(x);
12179 #endif
12180 void _print(ll x){cerr<<x;}
12181 void _print(string x){cerr<<x;}
12182 mt19937 rng(chrono::steady_clock::now().time_since_epoch().count());
12183 template<class T,class V> void _print(pair<T,V> p) {cerr<<"{";
12184 _print(p.first);cerr<<","; _print(p.second);cerr<<"}";}
12185 template<class T>void _print(vector<T> v) {cerr<<" [ "; for (T i:v){_print(i);cerr<<" ";}cerr<<"]";}
12186 template<class T>void _print(set<T> v) {cerr<<" [ "; for (T i:v){_print(i); cerr<<" ";}cerr<<"]";}
12187 template<class T>void _print(multiset<T> v) {cerr<<" [ "; for (T i:v){_print(i);cerr<<" ";}cerr<<"]";}
12188 template<class T,class V>void _print(map<T, V> v) {cerr<<" [ "; for(auto i:v){_print(i);cerr<<" ";} cerr<<"]";}
12189 template<class T> using oset=tree<T, null_type, less<T>, rb_tree_tag, tree_order_statistics_node_update>;
12190 template<class T> using muloset=tree<T, null_type, less_equal<T>, rb_tree_tag, tree_order_statistics_node_update>;
12191 //-----
12192 -----
12193 const ll MOD=1e9+7;
12194 const ll MAX=500500;
12195 void solve(){
12196     ll n,x; cin>>n>>x;
12197     ll ans=0;

```

```

12196     while(x) {
12197         ll cur=min(x,n);
12198         if((cur&1) != (x&1)) {
12199             cur--;
12200         }
12201         ans+=cur;
12202         x=(x-cur)/2;
12203     }
12204     cout<<ans<<endl;
12205     return;
12206 }
12207
12208 //MAXIMISEBITS-EDITOR
12209 #include "bits/stdc++.h"
12210 using namespace std;
12211 #define ll long long
12212 #define pb push_back
12213 #define all(_obj) _obj.begin(), _obj.end()
12214 #define F first
12215 #define S second
12216 #define pll pair<ll, ll>
12217 #define vll vector<ll>
12218 ll INF = 1e18;
12219 const int N = 1e5 + 11, mod = 1e9 + 7;
12220 ll max(ll a, ll b) { return ((a > b) ? a : b); }
12221 ll min(ll a, ll b) { return ((a > b) ? b : a); }
12222 mt19937 rng(chrono::steady_clock::now().time_since_epoch().count());
12223 void sol(void)
12224 {
12225     int n, k;
12226     cin >> n >> k;
12227     ll ans = 0;
12228     for (int i = 0; i < 30; i++)
12229     {
12230         if (n % 2 == k % 2)
12231         {
12232             ans += min(n, k);
12233             k -= min(n, k);
12234         }
12235         else
12236         {
12237             ans += min((n - 1), k);
12238             k -= min(n - 1, k);
12239         }
12240         k /= 2;
12241     }
12242     cout << ans << '\n';
12243     return;
12244 }
12245 int main()
12246 {
12247     ios_base::sync_with_stdio(false);
12248     cin.tie(NULL), cout.tie(NULL);
12249     int test = 1;
12250     cin >> test;
12251     while (test--)
12252         sol();
12253 }
12254
12255 //MAXSCORE
12256 #include <bits/stdc++.h>
12257 #include <ext/pb_ds/tree_policy.hpp>
12258 #include <ext/pb_ds/assoc_container.hpp>
12259 using namespace __gnu_pbds;
12260 using namespace std;
12261 #define ll long long
12262 const ll INF_MUL=1e13;
12263 const ll INF_ADD=1e18;
12264 #define pb push_back

```

```

12265 #define mp make_pair
12266 #define nline "\n"
12267 #define f first
12268 #define s second
12269 #define pll pair<ll,ll>
12270 #define vl vector<ll>
12271 #define vvl vector<vector<ll>>
12272 #define vvvl vector<vector<vector<ll>>>
12273 #define all(v) v.begin(),v.end()
12274 #ifndef ONLINE_JUDGE
12275 #define debug(x) cerr<<#x<<" "; _print(x); cerr<<nline;
12276 #else
12277 #define debug(x);
12278 #endif
12279 void _print(ll x){cerr<<x;}
12280 void _print(string x){cerr<<x;}
12281 mt19937 rng(chrono::steady_clock::now().time_since_epoch().count());
12282 template<class T,class V> void _print(pair<T,V> p) {cerr<<"{";
12283 _print(p.first);cerr<<","; _print(p.second);cerr<<"}";}
12284 template<class T>void _print(vector<T> v) {cerr<<" [ "; for (T i:v){_print(i);cerr<<"
12285 ";}cerr<<"]";}
12286 template<class T>void _print(set<T> v) {cerr<<" [ "; for (T i:v){_print(i); cerr<<"
12287 ";}cerr<<"]";}
12288 template<class T>void _print(multiset<T> v) {cerr<<" [ "; for (T
12289 i:v){_print(i);cerr<<" ";}cerr<<"]";}
12290 template<class T,class V>void _print(map<T, V> v) {cerr<<" [ "; for(auto i:v)
12291 {_print(i);cerr<<" ";} cerr<<"]";}
12292 template<class T> using oset=tree<T, null_type, less<T>, rb_tree_tag,
12293 tree_order_statistics_node_update>;
12294 template<class T> using muloset=tree<T, null_type, less_equal<T>, rb_tree_tag,
12295 tree_order_statistics_node_update>;
12296 //-----
12297 -----
12298
12299 const ll MOD=1e9+7;
12300 const ll MAX=500500;
12301 vector<ll> value(MAX);
12302 vector<vector<ll>> adj;
12303 vector<multiset<ll>> track(MAX);
12304 void dfs(ll cur,ll par){
12305     for(auto chld:adj[cur]){
12306         if(chld!=par){
12307             dfs(chld,cur);
12308             if(track[chld].size()>track[cur].size()){
12309                 swap(track[chld],track[cur]);
12310             }
12311             for(auto i:track[chld]){
12312                 track[cur].insert(i);
12313             }
12314         }
12315     }
12316     if(track[cur].empty()){
12317         track[cur].insert(value[cur]);
12318     }
12319     else{
12320         auto fs=track[cur].begin();
12321         ll val=*fs;
12322         if(val<value[cur]){
12323             track[cur].erase(fs);
12324             track[cur].insert(value[cur]);
12325         }
12326     }
12327 }
12328 void solve(){
12329     ll n; cin>>n;
12330     ll sum=0;
12331     vector<pair<ll,pair<ll,ll>>> segments;
12332     adj.clear(); adj.resize(2*n+5);
12333     for(ll i=1;i<=n;i++){

```

```

12325         ll l,r,v; cin>>l>>r>>v;
12326         sum+=v;
12327         segments.pb({l,{-r,v}});
12328         segments.pb({i,{-i,0}});
12329     }
12330     sort(all(segments));
12331     ll pos=2;
12332     vector<pair<ll,ll>> prv; prv.pb({n,1});
12333     value[1]=0;
12334     for(auto it:segments){
12335         while(!prv.empty()){
12336             auto i=prv.back();
12337             if(i.f<it.f){
12338                 prv.pop_back();
12339             }
12340             else{
12341                 break;
12342             }
12343         }
12344         ll par=(prv.back()).s;
12345         adj[par].pb(pos);
12346         value[pos]=it.s.s;
12347         prv.pb({-it.s.f,pos++});
12348     }
12349     dfs(1,0);
12350     ll ans=0;
12351     for(auto it:track[1]){
12352         ans+=it;
12353     }
12354     for(ll i=1;i<=2*n;i++){
12355         track[i].clear();
12356     }
12357     cout<<ans<<endl;
12358     cerr<<sum<<endl;
12359     return;
12360 }
12361 int main()
12362 {
12363     ios_base::sync_with_stdio(false);
12364     cin.tie(NULL);
12365     #ifndef ONLINE_JUDGE
12366     freopen("input.txt", "r", stdin);
12367     freopen("output.txt", "w", stdout);
12368     freopen("error.txt", "w", stderr);
12369     #endif
12370     ll test_cases=1;
12371     cin>>test_cases;
12372     while(test_cases--){
12373         solve();
12374     }
12375     cout<<fixed<<setprecision(15);
12376     cerr<<"Time:"<<1000*((double)clock())/(double)CLOCKS_PER_SEC<<"ms\n";
12377 }
12378
12379 //SORTRRAY
12380 #include <bits/stdc++.h>
12381 using namespace std;
12382
12383
12384 /*
12385 -----Input Checker-----
12386 */
12387
12388 long long readInt(long long l,long long r,char endd){
12389     long long x=0;
12390     int cnt=0;
12391     int fi=-1;
12392     bool is_neg=false;
12393     while(true){

```

```

12394     char g=getchar();
12395     if(g=='-'){
12396         assert(fi==-1);
12397         is_neg=true;
12398         continue;
12399     }
12400     if('0'<=g && g<='9'){
12401         x*=10;
12402         x+=g-'0';
12403         if(cnt==0){
12404             fi=g-'0';
12405         }
12406         cnt++;
12407         assert(fi!=0 || cnt==1);
12408         assert(fi!=0 || is_neg==false);
12409
12410         assert(!(cnt>19 || ( cnt==19 && fi>1) ));
12411     } else if(g==endd){
12412         if(is_neg){
12413             x= -x;
12414         }
12415
12416         if(!(l <= x && x <= r))
12417         {
12418             cerr << l << ' ' << r << ' ' << x << '\n';
12419             assert(1 == 0);
12420         }
12421
12422         return x;
12423     } else {
12424         assert(false);
12425     }
12426 }
12427 }
12428 string readString(int l,int r,char endd){
12429     string ret="";
12430     int cnt=0;
12431     while(true){
12432         char g=getchar();
12433         assert(g!=-1);
12434         if(g==endd){
12435             break;
12436         }
12437         cnt++;
12438         ret+=g;
12439     }
12440     assert(l<=cnt && cnt<=r);
12441     return ret;
12442 }
12443 long long readIntSp(long long l,long long r){
12444     return readInt(l,r,' ');
12445 }
12446 long long readIntLn(long long l,long long r){
12447     return readInt(l,r,'\n');
12448 }
12449 string readStringLn(int l,int r){
12450     return readString(l,r,'\n');
12451 }
12452 string readStringSp(int l,int r){
12453     return readString(l,r,' ');
12454 }
12455
12456
12457 /*
12458 -----Main code starts here-----
12459 */
12460
12461 const int MAX_T = 1e5;
12462 const int MAX_N = 1e5;

```

```

12463 const int MAX_SUM_LEN = 1e5;
12464
12465 #define fast ios_base::sync_with_stdio(0); cin.tie(0); cout.tie(0)
12466 #define ff first
12467 #define ss second
12468 #define mp make_pair
12469 #define ll long long
12470 #define rep(i,n) for(int i=0;i<n;i++)
12471 #define rev(i,n) for(int i=n;i>=0;i--)
12472 #define rep_a(i,a,n) for(int i=a;i<n;i++)
12473 #define pb push_back
12474
12475 int sum_n = 0, sum_m = 0;
12476 int max_n = 0, max_m = 0;
12477 int yess = 0;
12478 int nos = 0;
12479 int total_ops = 0;
12480 ll mod = 1000000007;
12481
12482 using ii = pair<ll,ll>;
12483
12484
12485 void solve(){
12486
12487     int n = readIntLn(2, 1e5);
12488     sum_n+=n;
12489     int a[n];
12490
12491     rep(i,n){
12492         if(i<n-1) a[i] = readIntSp(1,1e5);
12493         else a[i] = readIntLn(1,1e5);
12494     }
12495
12496     int b[n];
12497
12498     rep(i,n){
12499         b[i] = 1e5+i-a[i];
12500     }
12501
12502     vector<vector<int> > v(20);
12503
12504     int q = 0;
12505
12506     rep(i,20){
12507         rep(j,n){
12508             if((b[j]>>i)&1) v[i].pb(j);
12509         }
12510         if(!v[i].empty()) q++;
12511     }
12512
12513     cout<<q<<'\n';
12514     rep(i,20){
12515         if(!v[i].empty()){
12516             cout<<v[i].size()<<" "<<(1<<i)<<'\n';
12517             for(auto h:v[i]) cout<<h+1<<" ";
12518             cout<<'\n';
12519         }
12520     }
12521 }
12522
12523
12524 signed main()
12525 {
12526
12527     #ifndef ONLINE_JUDGE
12528     freopen("input.txt", "r" , stdin);
12529     freopen("output.txt", "w" , stdout);
12530     #endif
12531     fast;

```



```

12532
12533     int t = 1;
12534
12535     t = readIntLn(1,5e4);
12536
12537     for(int i=1;i<=t;i++)
12538     {
12539         solve();
12540     }
12541
12542     assert(getchar() == -1);
12543     assert(sum_n<=1e5);
12544
12545     cerr<<"SUCCESS\n";
12546     cerr<<"Tests : " << t << '\n';
12547     cerr<<"Sum of lengths : " << sum_n <<" "<<sum_m<<'\n';
12548     // cerr<<"Maximum length : " << max_n <<'\n';
12549     // // cerr<<"Total operations : " << total_ops << '\n';
12550     // cerr<<"Answered yes : " << yess << '\n';
12551     // cerr<<"Answered no : " << nos << '\n';
12552
12553     cerr << "Time : " << 1000 * ((double)clock()) / (double)CLOCKS_PER_SEC << "ms\n";
12554 }
12555
12556 //SORTRRAY-EDITOR
12557 #include "bits/stdc++.h"
12558 using namespace std;
12559 #define ll long long
12560 #define pb push_back
12561 #define all(_obj) _obj.begin(), _obj.end()
12562 #define F first
12563 #define S second
12564 #define pll pair<ll, ll>
12565 #define vll vector<ll>
12566 ll INF = 1e18;
12567 const int N = 1e5 + 11, mod = 1e9 + 7;
12568 ll max(ll a, ll b) { return ((a > b) ? a : b); }
12569 ll min(ll a, ll b) { return ((a > b) ? b : a); }
12570 mt19937 rng(chrono::steady_clock::now().time_since_epoch().count());
12571 void sol(void)
12572 {
12573     int n;
12574     cin >> n;
12575     vll v(n);
12576     for (int i = 0; i < n; i++)
12577     {
12578         cin >> v[i];
12579         v[i] = 1e5 + i - v[i];
12580     }
12581     vector<vector<int>> ans;
12582     for (int i = 0; i < 20; i++)
12583     {
12584         vector<int> ids;
12585         for (int j = 0; j < n; j++)
12586         {
12587             if ((1 << i) & (v[j]))
12588                 ids.pb(j + 1);
12589         }
12590         if (ids.size() == 0)
12591             continue;
12592         vector<int> tempans;
12593         tempans.pb(ids.size());
12594         tempans.pb(1 << i);
12595         ans.pb(tempans);
12596         ans.pb(ids);
12597     }
12598     cout << ans.size()/2 << '\n';
12599     for (auto x : ans)
12600     {

```

```

12601         for (auto y : x)
12602             cout << y << ' ';
12603         cout << '\n';
12604     }
12605     return;
12606 }
12607 int main()
12608 {
12609     ios_base::sync_with_stdio(false);
12610     cin.tie(NULL), cout.tie(NULL);
12611     int test = 1;
12612     cin >> test;
12613     while (test--)
12614         sol();
12615 }
12616
12617 //MINORMAX
12618
12619 #include <bits/stdc++.h>
12620 #include <ext/pb_ds/tree_policy.hpp>
12621 #include <ext/pb_ds/assoc_container.hpp>
12622 using namespace __gnu_pbds;
12623 using namespace std;
12624 #define ll long long
12625 const ll INF_MUL=1e13;
12626 const ll INF_ADD=1e18;
12627 #define pb push_back
12628 #define mp make_pair
12629 #define nline "\n"
12630 #define f first
12631 #define s second
12632 #define pll pair<ll,ll>
12633 #define vl vector<ll>
12634 #define vvl vector<vector<ll>>
12635 #define vvvl vector<vector<vector<ll>>>
12636 #define all(v) v.begin(),v.end()
12637 #ifndef ONLINE_JUDGE
12638 #define debug(x) cerr<<#x<<" "; _print(x); cerr<<nline;
12639 #else
12640 #define debug(x);
12641 #endif
12642 void _print(ll x){cerr<<x;}
12643 void _print(string x){cerr<<x;}
12644 mt19937 rng(chrono::steady_clock::now().time_since_epoch().count());
12645 template<class T,class V> void _print(pair<T,V> p) {cerr<<"{";
12646     _print(p.first);cerr<<","; _print(p.second);cerr<<"}";}
12647 template<class T>void _print(vector<T> v) {cerr<<" [ "; for (T i:v){_print(i);cerr<<"
12648     ";}cerr<<"]";}
12649 template<class T>void _print(set<T> v) {cerr<<" [ "; for (T i:v){_print(i); cerr<<"
12650     ";}cerr<<"]";}
12651 template<class T>void _print(multiset<T> v) {cerr<<" [ "; for (T
12652     i:v){_print(i);cerr<<" ";}cerr<<"]";}
12653 template<class T,class V>void _print(map<T, V> v) {cerr<<" [ "; for(auto i:v)
12654     {_print(i);cerr<<" ";} cerr<<"]";}
12655 template<class T> using oset=tree<T, null_type, less<T>, rb_tree_tag,
12656     tree_order_statistics_node_update>;
12657 template<class T> using muloset=tree<T, null_type, less_equal<T>, rb_tree_tag,
12658     tree_order_statistics_node_update>;
12659 //-----
12660 -----
12661
12662 const ll MOD=1e9+7;
12663 const ll MAX=500500;
12664 void solve(){
12665     ll n; cin>>n;
12666     vector<ll> a(n+5);
12667     for(ll i=1;i<=n;i++){
12668         cin>>a[i];
12669     }

```

```

12661     ll nax=0,nin=INF_ADD;
12662     for(ll i=1;i<=n;i++){
12663         nax=max(nax,a[i]);
12664         nin=min(nin,a[i]);
12665         if((nax!=a[i])&&(nin!=a[i])){
12666             cout<<"NO\n";
12667             return;
12668         }
12669     }
12670     cout<<"YES\n";
12671     return;
12672 }
12673 int main()
12674 {
12675     ios_base::sync_with_stdio(false);
12676     cin.tie(NULL);
12677     #ifndef ONLINE_JUDGE
12678     freopen("input.txt", "r", stdin);
12679     freopen("output.txt", "w", stdout);
12680     freopen("error.txt", "w", stderr);
12681     #endif
12682     ll test_cases=1;
12683     cin>>test_cases;
12684     while(test_cases--){
12685         solve();
12686     }
12687     cout<<fixed<<setprecision(15);
12688     cerr<<"Time:"<<1000*((double)clock())/(double)CLOCKS_PER_SEC<<"ms\n";
12689 }
12690
12691 //MINORMAX-EDITOR
12692 #include "bits/stdc++.h"
12693 using namespace std;
12694 #define ll long long
12695 #define pb push_back
12696 #define all(_obj) _obj.begin(), _obj.end()
12697 #define F first
12698 #define S second
12699 #define pll pair<ll, ll>
12700 #define vll vector<ll>
12701 ll INF = 1e18;
12702 const int N = 1e5 + 11, mod = 1e9 + 7;
12703 ll max(ll a, ll b) { return ((a > b) ? a : b); }
12704 ll min(ll a, ll b) { return ((a > b) ? b : a); }
12705 mt19937 rng(chrono::steady_clock::now().time_since_epoch().count());
12706 void sol(void)
12707 {
12708     int n, lastmin, lastmax;
12709     cin >> n;
12710     vll v(n);
12711     bool flag = true;
12712     for (int i = 0; i < n; i++)
12713         cin >> v[i];
12714     lastmax = lastmin = v[0];
12715     for (int i = 1; i < n; i++)
12716     {
12717         if (v[i] >= lastmax)
12718             lastmax = v[i];
12719         else if (v[i] <= lastmin)
12720             lastmin = v[i];
12721         else
12722             flag = false;
12723     }
12724     if (flag)
12725         cout << "YES\n";
12726     else
12727         cout << "NO\n";
12728     return;
12729 }

```

```

12730 int main()
12731 {
12732     ios_base::sync_with_stdio(false);
12733     cin.tie(NULL), cout.tie(NULL);
12734     int test = 1;
12735     cin >> test;
12736     while (test--)
12737         sol();
12738 }
12739
12740 //WBLACK
12741
12742 #include <bits/stdc++.h>
12743
12744
12745 using namespace std;
12746
12747 #include <ext/pb_ds/assoc_container.hpp>
12748 #include <ext/pb_ds/tree_policy.hpp>
12749 using namespace __gnu_pbds;
12750
12751 #define ordered_set tree<ll, null_type, less_equal<ll>, rb_tree_tag,
tree_order_statistics_node_update>
12752
12753 #define ll long long int
12754
12755 #define ld long double
12756 #define forn(i, x, n) for (ll i = x; i < n; i++)
12757 #define fornbn(i, n, x) for (ll i = n; i >=x; i--)
12758 #define all(x) x.begin(), x.end()
12759 #define pii pair<ll, ll>
12760 #define MOD 1000000007
12761 #define MAX 300007
12762 #define endl "\n" // REMOVE in lleraction problem
12763 #define debug cout << "K"
12764 vector<ll> visited(MAX), color(MAX), dist(MAX, -1);
12765 vector<ll> graph[MAX];
12766 vector<ll> parent(MAX);
12767 vector<pii> graph2[MAX];
12768
12769
12770 vector<ll> A(MAX) , B(MAX);
12771 ll dp[MAX][3];
12772 ll visited2[MAX][3];
12773 //state 0-> black , 1->white , 2->none
12774 ll dfs(ll node , ll state , ll p)
12775 {
12776
12777
12778     ll val =A[node];
12779     if(state!=2)
12780         val =state;
12781     if((state!=2 && val!=B[node]) || (state==2 && A[node]!=B[node]))
12782     {
12783         dp[node][state] = LLONG_MAX;
12784         return LLONG_MAX;
12785     }
12786     visited2[node][state] =1;
12787     ll ans = 0;
12788     if(state==1)
12789     {
12790         for(auto child : graph[node])
12791         {
12792             ll tmp = LLONG_MAX;
12793             if(child==p)
12794                 continue;
12795             if(B[child]==0)
12796             {
12797                 if(visited2[child][0]){}

```

```

12798         else
12799             dp[child][0] = dfs(child , 0 , node);
12800             tmp = min(tmp ,dp[child][0]);
12801     }
12802     if(B[child]==1)
12803     {
12804         if(visited2[child][1]==1){}
12805         else
12806             dp[child][1] = dfs(child , 1 , node);
12807             tmp = min(tmp ,dp[child][1] -1);
12808     }
12809     if(tmp!=LLONG_MAX)
12810         ans+=tmp;
12811 }
12812 dp[node][state] = ans+1;
12813 return dp[node][state];
12814 }
12815 else
12816 if(state==0){
12817 for(auto child : graph[node])
12818 {
12819     ll tmp = LLONG_MAX;
12820
12821     if(child==p)
12822         continue;
12823     if(B[child]==0)
12824     {
12825         if(visited2[child][0]==1){}
12826         else
12827             dp[child][0] = dfs(child , 0 , node);
12828             tmp = min(tmp ,dp[child][0]-1);
12829     }
12830     if(B[child]==1)
12831     {
12832         if(visited2[child][1]==1){}
12833         else
12834             dp[child][1] = dfs(child , 1 , node);
12835             tmp = min(tmp ,dp[child][1]);
12836     }
12837 }
12838
12839     if(tmp!=LLONG_MAX)
12840         ans+=tmp;
12841 }
12842 dp[node][state] = ans+1;
12843 return dp[node][state];
12844 }
12845 else
12846 {
12847     if(A[node]!=B[node])
12848     {
12849         dp[node][2] = LLONG_MAX;
12850         return dp[node][2];
12851     }
12852     for(auto child : graph[node])
12853     {
12854         ll tmp = LLONG_MAX;
12855         if(child==p){
12856             continue;
12857         }
12858         if(B[child]==0 && A[child]==1)
12859         {
12860             ll a = LLONG_MAX;
12861             if(visited2[child][0]==1){
12862                 a = dp[child][0];
12863             }
12864             else
12865             {
12866                 dp[child][0] = dfs(child , 0 , node);

```

```

12867         a = dp[child][0];
12868     }
12869     tmp = min(tmp ,a);
12870 }
12871 if(B[child]==1 && A[child] == 0)
12872 {
12873     ll a = LLONG_MAX;
12874     if(visited2[child][1]==1){
12875         a = dp[child][1];
12876     }
12877     else{
12878         dp[child][1] = dfs(child , 1 , node );
12879         a = dp[child][1];
12880     }
12881     tmp = min(tmp ,a);
12882 }
12883 if(B[child]==A[child])
12884 {
12885     ll a = LLONG_MAX;
12886     if(visited2[child][2]==1){
12887     }
12888     else
12889         dp[child][2] = dfs(child , 2 , node);
12890     tmp = min(tmp ,dp[child][2]);
12891     if(visited2[child][B[child]]==1){
12892         a = dp[child][B[child]];
12893     }
12894     else{
12895         dp[child][B[child]] = dfs(child , B[child] , node);
12896         a = dp[child][B[child]];
12897     }
12898     tmp = min(tmp ,a);
12899 }
12900 if(tmp!=LLONG_MAX)
12901     ans+=tmp;
12902 }
12903 return dp[node][state] = ans;
12904 }
12905 }
12906
12907 int main()
12908 {
12909
12910     ios_base::sync_with_stdio(false);
12911     cin.tie(NULL);
12912     ll t=1;
12913     cin>>t;
12914     while (t--)
12915     {
12916         ll n;
12917         cin>>n;
12918
12919         forn(i , 0 , n)
12920             cin>>A[i];
12921         forn(i, 0, n)
12922             cin>>B[i];
12923         forn(i ,0,n-1)
12924         {
12925             ll a , b;
12926             cin>>a>>b;
12927             a--;
12928             b--;
12929
12930             graph[a].push_back(b);
12931             graph[b].push_back(a);
12932         }
12933         forn(i ,0, n+1)
12934             forn(j ,0 ,3)
12935         {

```

```

12936         dp[i][j] = LLONG_MAX;
12937         visited2[i][j] = 0;}
12938
12939         dfs(0 ,0 ,0 ); dfs(0 , 1 , 0); dfs(0 , 2 ,0);
12940         cout<<min(dp[0][1], min(dp[0][0] ,dp[0][2]))<<endl;
12941         forn(i, 0 ,n+1)
12942             graph[i].clear();
12943     }
12944 }
12945
12946 //WBLACK-EDITOR
12947
12948 #include "bits/stdc++.h"
12949 using namespace std;
12950 #define ll long long
12951 #define pb push_back
12952 #define all(_obj) _obj.begin(), _obj.end()
12953 #define F first
12954 #define S second
12955 #define pll pair<ll, ll>
12956 #define vll vector<ll>
12957 ll INF = 1e18;
12958 const int N = 3e5 + 11, mod = 1e9 + 7;
12959 ll max(ll a, ll b) { return ((a > b) ? a : b); }
12960 ll min(ll a, ll b) { return ((a > b) ? b : a); }
12961 mt19937 rng(chrono::steady_clock::now().time_since_epoch().count());
12962 int a[N], b[N];
12963 long long black[N], white[N], none[N];
12964 vll v[N];
12965 void dfs(int u, int p)
12966 {
12967     if (b[u])
12968         white[u]++;
12969     else
12970         black[u]++;
12971     for (auto x : v[u])
12972     {
12973         if (x == p)
12974             continue;
12975         dfs(x, u);
12976         none[u] += min(min(none[x], 1 + white[x]), 1 + black[x]);
12977         if (b[u])
12978         {
12979             black[u] += black[x];
12980             white[u] += black[x];
12981         }
12982         if (!b[u])
12983         {
12984             white[u] += white[x];
12985             black[u] += white[x];
12986         }
12987     }
12988     if (a[u] != b[u])
12989         none[u] = 1e9;
12990     return;
12991 }
12992 void sol(void)
12993 {
12994     int n;
12995     cin >> n;
12996     for (int i = 1; i <= n; i++)
12997         cin >> a[i], v[i].clear(), black[i] = white[i] = none[i] = 0;
12998     for (int i = 1; i <= n; i++)
12999         cin >> b[i];
13000     for (int i = 1; i <= n - 1; i++)
13001     {
13002         int x, y;
13003         cin >> x >> y;
13004         v[x].pb(y);

```

```

13005         v[y].pb(x);
13006     }
13007     dfs(1, -1);
13008     cout << min(min(1 + black[1], none[1]), 1 + white[1]) << '\n';
13009     return;
13010 }
13011 int main()
13012 {
13013     ios_base::sync_with_stdio(false);
13014     cin.tie(NULL), cout.tie(NULL);
13015     int test = 1;
13016     cin >> test;
13017     while (test--)
13018         sol();
13019 }
13020
13021 //DEARRANGE
13022 //Utkarsh.25dec
13023 #include <bits/stdc++.h>
13024 #define ll long long int
13025 #define pb push_back
13026 #define mp make_pair
13027 #define mod 1000000007
13028 #define vl vector <ll>
13029 #define all(c) (c).begin(), (c).end()
13030 using namespace std;
13031 ll power(ll a, ll b) {ll res=1; a%=mod; assert(b>=0);
13032 for(; b; b>>=1) {if(b&1) res=res*a%mod; a=a*a%mod;} return res;}
13033 ll modInverse(ll a) {return power(a, mod-2);}
13034 const int N=500023;
13035 bool vis[N];
13036 vector <int> adj[N];
13037 long long readInt(long long l, long long r, char endd) {
13038     long long x=0;
13039     int cnt=0;
13040     int fi=-1;
13041     bool is_neg=false;
13042     while(true){
13043         char g=getchar();
13044         if(g=='-'){
13045             assert(fi==-1);
13046             is_neg=true;
13047             continue;
13048         }
13049         if('0'<=g && g<='9'){
13050             x*=10;
13051             x+=g-'0';
13052             if(cnt==0){
13053                 fi=g-'0';
13054             }
13055             cnt++;
13056             assert(fi!=0 || cnt==1);
13057             assert(fi!=0 || is_neg==false);
13058
13059             assert(!(cnt>19 || ( cnt==19 && fi>1) ));
13060         } else if(g==endd){
13061             if(is_neg){
13062                 x= -x;
13063             }
13064
13065             if(!(l <= x && x <= r))
13066             {
13067                 cerr << l << ' ' << r << ' ' << x << '\n';
13068                 assert(1 == 0);
13069             }
13070
13071             return x;
13072         } else {
13073             assert(false);

```



```

13073     }
13074 }
13075 }
13076 string readString(int l,int r,char endd){
13077     string ret="";
13078     int cnt=0;
13079     while(true){
13080         char g=getchar();
13081         assert(g!=-1);
13082         if(g==endd){
13083             break;
13084         }
13085         cnt++;
13086         ret+=g;
13087     }
13088     assert(l<=cnt && cnt<=r);
13089     return ret;
13090 }
13091 long long readIntSp(long long l,long long r){
13092     return readInt(l,r,' ');
13093 }
13094 long long readIntLn(long long l,long long r){
13095     return readInt(l,r,'\n');
13096 }
13097 string readStringLn(int l,int r){
13098     return readString(l,r,'\n');
13099 }
13100 string readStringSp(int l,int r){
13101     return readString(l,r,' ');
13102 }
13103 mt19937 rng(chrono::steady_clock::now().time_since_epoch().count());
13104 ll myrand(ll l,ll r)
13105 {
13106     ll temp=rng();
13107     temp=abs(temp);
13108     temp%=(r-l+1);
13109     temp+=l;
13110     return temp;
13111 }
13112 int sumN=0;
13113 void solve()
13114 {
13115     int n=readInt(3,1000,'\n');
13116     sumN+=n;
13117     assert(sumN<=1000);
13118     int A[n+1]={0};
13119     int mark[n+1]={0};
13120     for(int i=1;i<=n;i++)
13121     {
13122         if(i==n)
13123             A[i]=readInt(1,n,'\n');
13124         else
13125             A[i]=readInt(1,n,' ');
13126         mark[A[i]]=1;
13127     }
13128     for(int i=1;i<=n;i++)
13129         assert(mark[i]==1);
13130     int l=n+1,r=0;
13131     for(int i=1;i<=n;i++)
13132     {
13133         if(A[i]!=i)
13134             r=i;
13135     }
13136     for(int i=n;i>=1;i--)
13137     {
13138         if(A[i]!=i)
13139             l=i;
13140     }
13141     if(l>r)

```

```

13142 {
13143     cout<<0<<'\n';
13144     return;
13145 }
13146 for(int i=1;i<=r;i++)
13147 {
13148     if(A[i]==i)
13149     {
13150         cout<<2<<'\n';
13151         cout<<1<<' '<<n<<'\n';
13152         vector <int> v;
13153         for(int i=1;i<=n;i++)
13154             v.pb(i);
13155         int iter=0;
13156         while(true)
13157         {
13158             // iter++;
13159             // if(iter>=100)
13160             // {
13161             //     cout<<"BAD\n";
13162             //     cout<<n<<'\n';
13163             //     for(int i=1;i<=n;i++)
13164             //         cout<<A[i]<<' ';
13165             //     cout<<'\n';
13166             //     exit(0);
13167             // }
13168             shuffle(all(v),rng);
13169             int flag=1;
13170             for(int i=0;i<n;i++)
13171             {
13172                 if(i==0)
13173                 {
13174                     if(v[i]==A[i+1])
13175                     {
13176                         flag=0;
13177                         break;
13178                     }
13179                     else
13180                         continue;
13181                 }
13182                 if(v[i]==i+1 || v[i]==A[i+1])
13183                 {
13184                     flag=0;
13185                     break;
13186                 }
13187             }
13188             if(flag==1)
13189                 break;
13190         }
13191         for(auto it:v)
13192             cout<<it<<' ';
13193         cout<<'\n';
13194         if(v[0]!=1)
13195             cout<<1<<' '<<n<<'\n';
13196         else
13197             cout<<2<<' '<<n<<'\n';
13198         for(int i=1;i<=n;i++)
13199             cout<<i<<' ';
13200         cout<<'\n';
13201         return;
13202     }
13203 }
13204 cout<<1<<'\n';
13205 cout<<1<<' '<<r<<'\n';
13206 for(int i=1;i<=n;i++)
13207     cout<<i<<' ';
13208 cout<<'\n';
13209 }
13210 int main()

```

```

13211 {
13212     #ifndef ONLINE_JUDGE
13213     freopen("input.txt", "r", stdin);
13214     freopen("output.txt", "w", stdout);
13215     #endif
13216     ios_base::sync_with_stdio(false);
13217     cin.tie(NULL), cout.tie(NULL);
13218     int T=readInt(1,200,'\n');
13219     while(T--)
13220         solve();
13221     assert(getchar() == -1);
13222     cerr << "Time : " << 1000 * ((double)clock()) / (double)CLOCKS_PER_SEC << "ms\n";
13223 }
13224
13225 //DEARRANGE-EDITOR
13226
13227 #include "bits/stdc++.h"
13228 using namespace std;
13229 #define ll long long
13230 #define pb push_back
13231 #define all(_obj) _obj.begin(), _obj.end()
13232 #define F first
13233 #define S second
13234 #define pll pair<ll, ll>
13235 #define vll vector<ll>
13236 ll INF = 1e18;
13237 const int N = 1e5 + 11, mod = 1e9 + 7;
13238 ll max(ll a, ll b) { return ((a > b) ? a : b); }
13239 ll min(ll a, ll b) { return ((a > b) ? b : a); }
13240 mt19937 rng(chrono::steady_clock::now().time_since_epoch().count());
13241 void sol(void)
13242 {
13243     int n;
13244     cin >> n;
13245     vll v(n), vv(n);
13246     vll check1(3);
13247     check1 = {3, 2, 1};
13248     for (int i = 0; i < n; i++)
13249     {
13250         cin >> v[i];
13251         vv[i] = v[i];
13252     }
13253     if (v == check1)
13254     {
13255         cout << 2 << ' ';
13256         cout << 1 << ' ' << 2 << '\n';
13257         cout << 2 << ' ' << 3 << ' ' << 1 << '\n';
13258         cout << 1 << ' ' << 3 << '\n';
13259         cout << 1 << ' ' << 2 << ' ' << 3 << '\n';
13260         return;
13261     }
13262     //----- Case when answer is 0
13263     sort(all(vv));
13264     if (vv == v)
13265     {
13266         cout << 0 << '\n';
13267         return;
13268     }
13269     //----- Case when answer is 1
13270     int issortedtill[n], issortedfrom[n];
13271     if (v[0] == 1)
13272         issortedtill[0] = true;
13273     else
13274         issortedtill[0] = false;
13275     for (int i = 1; i < n; i++)
13276         if (issortedtill[i - 1] && v[i] == i + 1)
13277             issortedtill[i] = true;
13278     else
13279         issortedtill[i] = false;

```

```

13280     if (v[n - 1] == n)
13281         issortedfrom[n - 1] = true;
13282     else
13283         issortedfrom[n - 1] = false;
13284     for (int i = n - 2; i >= 0; i--)
13285         if (issortedfrom[i + 1] && v[i] == i + 1)
13286             issortedfrom[i] = true;
13287         else
13288             issortedfrom[i] = false;
13289     for (int i = 0; i < n; i++)
13290     {
13291         bool flag = true;
13292         for (int j = i; j < n; j++)
13293         {
13294             if (v[j] == j + 1)
13295                 flag = false;
13296             if (flag && (j == n - 1 || issortedfrom[j + 1]) && (!i || issortedtill[i - 1]))
13297             {
13298                 cout << 1 << '\n';
13299                 cout << i + 1 << ' ' << j + 1 << '\n';
13300                 for (auto x : vv)
13301                     cout << x << ' ';
13302                 cout << '\n';
13303                 return;
13304             }
13305         }
13306     }
13307     //----- Case when answer is 2
13308     cout << 2 << '\n';
13309     vector<int> tillnow;
13310     for (int i = 0; i < n; i)
13311     {
13312         vector<int> temp;
13313         if (n - i < 8)
13314             for (int j = i; j < n; j++)
13315                 temp.pb(v[j]);
13316         else
13317             for (int j = i; j <= i + 3; j++)
13318                 temp.pb(v[j]);
13319         sort(all(temp));
13320         bool flag = true;
13321         do
13322         {
13323             flag = true;
13324             for (int j = 0; j < temp.size(); j++)
13325             {
13326                 if (v[i + j] == temp[j] || temp[j] == i + j + 1)
13327                     flag = false;
13328             }
13329             if (flag)
13330                 break;
13331         } while (next_permutation(all(temp)));
13332         for (auto x : temp)
13333             tillnow.pb(x);
13334         i += temp.size();
13335     }
13336     cout << 1 << ' ' << n << '\n';
13337     for (auto x : tillnow)
13338         cout << x << ' ';
13339     cout << '\n';
13340     cout << 1 << ' ' << n << '\n';
13341     for (int i = 1; i <= n; i++)
13342         cout << i << ' ';
13343     return;
13344 }
13345
13346 //REMOVEADD
13347 #include <bits/stdc++.h>

```

```

13348 using namespace std;
13349 #define ll long long int
13350 #define FOR(i, x, n) for (ll i = x; i < n; i++)
13351 int main()
13352 {
13353     ios_base::sync_with_stdio(false);
13354     cin.tie(NULL);
13355     ll t=1;
13356     cin>>t;
13357     while (t--)
13358     {
13359         ll n , k;
13360         cin>>n;
13361         vector<ll> A(n);
13362         FOR(i,0 ,n)
13363             cin>>A[i];
13364         unordered_set<ll> st;
13365         ll ind = 0;
13366         ll ans =LLONG_MAX;
13367         FOR(i, 0 ,n)
13368         {
13369             if(st.count(A[i])==1)
13370             {
13371                 ll r = n-i;
13372                 ll l =ind;
13373                 // cout<<i<<"\n";
13374                 ans = min(ans , 2*min(l , r) + max(l , r));
13375                 FOR(j , ind , n)
13376                 {
13377                     if(A[i]==A[j])
13378                     {
13379                         ind = j+1;
13380                         break;
13381                     }
13382                     else
13383                         st.erase(A[j]);
13384                 }
13385                 st.insert(A[i]);
13386             }
13387             else
13388             {
13389                 st.insert(A[i]);
13390             }
13391         }
13392         ans = min(ind , ans);
13393         cout<<ans<<endl;
13394     }
13395 }
13396
13397
13398 //REMOVEADD-EDITOR
13399
13400 #include "bits/stdc++.h"
13401 using namespace std;
13402 #define ll long long
13403 #define pb push_back
13404 #define all(_obj) _obj.begin(), _obj.end()
13405 #define F first
13406 #define S second
13407 #define pll pair<ll, ll>
13408 #define vll vector<ll>
13409 ll INF = 1e18;
13410 const int N = 1e5 + 11, mod = 1e9 + 7;
13411 ll max(ll a, ll b) { return ((a > b) ? a : b); }
13412 ll min(ll a, ll b) { return ((a > b) ? b : a); }
13413 mt19937 rng(chrono::steady_clock::now().time_since_epoch().count());
13414 void sol(void)
13415 {
13416     int n;

```

```

13417     cin >> n;
13418     vll v(n);
13419     for (int i = 0; i < n; i++)
13420         cin >> v[i];
13421     int ans = 1e9;
13422     map<int, int> mp;
13423     int r = 0;
13424     for (int l = 0; l < n; l++)
13425     {
13426         while (r < n && !mp[v[r]])
13427         {
13428             mp[v[r]]++;
13429             r++;
13430         }
13431         ans = min(ans, 1 + 2 * (n - r));
13432         ans = min(ans, 2 * l + n - r);
13433         mp[v[l]]--;
13434     }
13435     cout << ans << '\n';
13436     return;
13437 }
13438 int main()
13439 {
13440     ios_base::sync_with_stdio(false);
13441     cin.tie(NULL), cout.tie(NULL);
13442     int test = 1;
13443     cin >> test;
13444     while (test--)
13445         sol();
13446 }
13447
13448 //PARFUN
13449 #include <bits/stdc++.h>
13450 #define maxn 111
13451 #define maxa 111
13452 #define ll long long int
13453 using namespace std;
13454
13455 bool dpo[maxn][maxn/2][2*maxa*maxn + 1], dpe[maxn][2*maxa*maxn + 1];
13456
13457 int main() {
13458     //freopen("inp18.in", "r", stdin);
13459     //freopen("inp18.out", "w", stdout);
13460     int t = 1;
13461     //cin >> t;
13462     while(t--) {
13463         int n;
13464         cin >> n;
13465         assert(n > 1 && n < 101);
13466         int offset = n*maxa;
13467         int a[n];
13468         for(int i = 0; i < n; i++) {
13469             cin >> a[i];
13470             assert(a[i] > 0 && a[i] < 101);
13471         }
13472         vector<int> odd, even;
13473         for(int i = 0; i < n; i++) {
13474             if(a[i]&1) odd.push_back(a[i]);
13475             else even.push_back(a[i]);
13476         }
13477         dpe[0][offset] = 1;
13478         for(int i = 0; i < even.size(); i++) {
13479             for(int j = 0; j <= 2*n*maxa; j++) {
13480                 if(j - even[i] >= 0)
13481                     dpe[i + 1][j - even[i]] |= dpe[i][j];
13482                 if(j + even[i] <= 2*n*maxa)
13483                     dpe[i + 1][j + even[i]] |= dpe[i][j];
13484             }
13485         }

```

```

13486     dpo[0][0][offset] = 1;
13487     for(int i = 0; i < odd.size(); i++) {
13488         for(int k = 0; k <= odd.size()/2; k++) {
13489             for(int j = 0; j <= 2*n*maxa; j++) {
13490                 if(j - odd[i] >= 0 && k < odd.size()/2)
13491                     dpo[i + 1][k + 1][j - odd[i]] |= dpo[i][k][j];
13492                 if(j + odd[i] <= 2*n*maxa)
13493                     dpo[i + 1][k][j + odd[i]] |= dpo[i][k][j];
13494             }
13495         }
13496     }
13497 }
13498 int esum = 0;
13499 for(int i = 0; i < even.size(); i++) esum += even[i];
13500 cerr << even.size() << " " << odd.size() << "\n";
13501 if(!odd.size()) {
13502     dpe[even.size()][esum + offset] = 0;
13503     dpe[even.size()][-esum + offset] = 0;
13504 } else if(odd.size() == 1) {
13505     dpe[even.size()][-esum + offset] = 0;
13506 }
13507 vector<int> epos, opos;
13508 for(int i = 0; i <= 2*n*maxa; i++) {
13509     if(dpe[even.size()][i])
13510         epos.push_back(i - offset);
13511 }
13512 for(int i = 0; i <= 2*n*maxa; i++) {
13513     if(dpo[odd.size()][odd.size()/2][i])
13514         opos.push_back(i - offset);
13515 }
13516 set<int> pos;
13517 for(int i = 0; i < epos.size(); i++)
13518     for(int j = 0; j < opos.size(); j++)
13519         pos.insert(epos[i] + opos[j]);
13520 cout << pos.size() << "\n";
13521 }
13522 }
13523
13524 //WINNERR
13525 //Utkarsh.25dec
13526 #include <bits/stdc++.h>
13527 #define ll long long int
13528 #define pb push_back
13529 #define mp make_pair
13530 #define mod 1000000007
13531 #define vl vector<ll>
13532 #define all(c) (c).begin(), (c).end()
13533 using namespace std;
13534 ll power(ll a, ll b) {ll res=1; a%=mod; assert(b>=0);
13535     for(; b; b>>=1) {if(b&1) res=res*a%mod; a=a*a%mod;} return res;}
13536 ll modInverse(ll a) {return power(a, mod-2);}
13537 const int N=500023;
13538 bool vis[N];
13539 vector<int> adj[N];
13540 long long readInt(long long l, long long r, char endd) {
13541     long long x=0;
13542     int cnt=0;
13543     int fi=-1;
13544     bool is_neg=false;
13545     while(true) {
13546         char g=getchar();
13547         if(g=='-') {
13548             assert(fi==-1);
13549             is_neg=true;
13550             continue;
13551         }
13552         if('0'<=g && g<='9') {
13553             x*=10;
13554             x+=g-'0';

```

```

13554         if(cnt==0){
13555             fi=g-'0';
13556         }
13557         cnt++;
13558         assert(fi!=0 || cnt==1);
13559         assert(fi!=0 || is_neg==false);
13560
13561         assert(!(cnt>19 || ( cnt==19 && fi>1) ));
13562     } else if(g==endd){
13563         if(is_neg){
13564             x= -x;
13565         }
13566
13567         if(!(l <= x && x <= r))
13568         {
13569             cerr << l << ' ' << r << ' ' << x << '\n';
13570             assert(1 == 0);
13571         }
13572
13573         return x;
13574     } else {
13575         assert(false);
13576     }
13577 }
13578 }
13579 string readString(int l,int r,char endd){
13580     string ret="";
13581     int cnt=0;
13582     while(true){
13583         char g=getchar();
13584         assert(g!=-1);
13585         if(g==endd){
13586             break;
13587         }
13588         cnt++;
13589         ret+=g;
13590     }
13591     assert(l<=cnt && cnt<=r);
13592     return ret;
13593 }
13594 long long readIntSp(long long l,long long r){
13595     return readInt(l,r,' ');
13596 }
13597 long long readIntLn(long long l,long long r){
13598     return readInt(l,r,'\n');
13599 }
13600 string readStringLn(int l,int r){
13601     return readString(l,r,'\n');
13602 }
13603 string readStringSp(int l,int r){
13604     return readString(l,r,' ');
13605 }
13606 void solve()
13607 {
13608     int PA,PB,QA,QB;
13609     PA=readInt(1,100,' ');
13610     PB=readInt(1,100,' ');
13611     QA=readInt(1,100,' ');
13612     QB=readInt(1,100,'\n');
13613     if (max (PA,PB)>max (QA,QB) )
13614         cout<<"Q\n";
13615     else if (max (PA,PB)<max (QA,QB) )
13616         cout<<"P\n";
13617     else
13618         cout<<"TIE\n";
13619 }
13620 int main()
13621 {
13622     #ifndef ONLINE_JUDGE

```



```

13623     freopen("input.txt", "r", stdin);
13624     freopen("output.txt", "w", stdout);
13625     #endif
13626     ios_base::sync_with_stdio(false);
13627     cin.tie(NULL), cout.tie(NULL);
13628     int T=readInt(1,1000,'\n');
13629     while(T--)
13630         solve();
13631     assert(getchar()==-1);
13632     cerr << "Time : " << 1000 * ((double)clock()) / (double)CLOCKS_PER_SEC << "ms\n";
13633 }
13634
13635 //WINNERR-EDITOR
13636 #include "bits/stdc++.h"
13637 using namespace std;
13638 #define ll long long
13639 #define all(_obj) _obj.begin(), _obj.end()
13640 #define F first
13641 #define S second
13642 #define pll pair<ll, ll>
13643 #define vll vector<ll>
13644 ll INF = 1e18;
13645 const int N = 1e5 + 11, mod = 1e9 + 7;
13646 ll max(ll a, ll b) { return ((a > b) ? a : b); }
13647 ll min(ll a, ll b) { return ((a > b) ? b : a); }
13648 mt19937 rng(chrono::steady_clock::now().time_since_epoch().count());
13649 void sol(void)
13650 {
13651     int pa, pb, qa, qb;
13652     cin >> pa >> pb >> qa >> qb;
13653     int x = max(pa, pb), y = max(qa, qb);
13654     if (x < y)
13655         cout << 'P' << '\n';
13656     else if (y < x)
13657         cout << 'Q' << '\n';
13658     else
13659         cout << "TIE\n";
13660     return;
13661 }
13662 int main()
13663 {
13664     ios_base::sync_with_stdio(false);
13665     cin.tie(NULL), cout.tie(NULL);
13666     int test = 1;
13667     cin >> test;
13668     while (test--)
13669         sol();
13670 }
13671
13672 //PARTHPAR
13673 #include "bits/stdc++.h"
13674 using namespace std;
13675 #define ll long long
13676 #define pb push_back
13677 #define all(_obj) _obj.begin(), _obj.end()
13678 #define F first
13679 #define S second
13680 #define pll pair<ll, ll>
13681 #define vll vector<ll>
13682 ll INF = 1e18;
13683 const int N = 1e5 + 11, mod = 1e9 + 7;
13684 ll max(ll a, ll b) { return ((a > b) ? a : b); }
13685 ll min(ll a, ll b) { return ((a > b) ? b : a); }
13686 mt19937 rng(chrono::steady_clock::now().time_since_epoch().count());
13687 void sol(void)
13688 {
13689     int n, k;
13690     cin >> n >> k;
13691     if (n % 2 == 0)

```

```

13692         cout << "YES\n";
13693     else if (k)
13694         cout << "YES\n";
13695     else
13696         cout << "NO\n";
13697     return;
13698 }
13699 int main()
13700 {
13701     ios_base::sync_with_stdio(false);
13702     cin.tie(NULL), cout.tie(NULL);
13703     int test = 1;
13704     cin >> test;
13705     while (test--)
13706         sol();
13707 }
13708
13709 //MAXMST
13710 #include<bits/stdc++.h>
13711 #define ll long long
13712 #define MOD 1000000007
13713 using namespace std;
13714 int main(){
13715     ll T;
13716     cin >> T;
13717     while(T--){
13718         ll n,m;
13719         cin >> n >> m;
13720         vector<ll>a(m);
13721         for(ll i=0;i<m;i++)cin >> a[i];
13722         sort(a.begin(),a.end());
13723         vector<bool>present(m,false);
13724         ll pos=0;
13725         ll c=1;
13726         ll ans=0;
13727         ll x=0;
13728         while(pos<m){
13729             present[pos]=true;
13730             if(x<n-1){
13731                 x++;
13732                 ans+=a[pos];
13733             }
13734             pos+=c;
13735             c++;
13736         }
13737         for(ll i=m-1;i>=0;i--){
13738             if(x==n-1)break;
13739             if(present[i])continue;
13740             x++;
13741             ans+=a[i];
13742         }
13743         cout << ans << endl;
13744     }
13745     return 0;
13746 }
13747
13748 //MAXMST-EDITOR
13749 #include "bits/stdc++.h"
13750 using namespace std;
13751 #define ll long long
13752 #define pb push_back
13753 #define all(_obj) _obj.begin(), _obj.end()
13754 #define F first
13755 #define S second
13756 #define pll pair<ll, ll>
13757 #define vll vector<ll>
13758 ll INF = 1e18;
13759 const int N = 1e5 + 11, mod = 1e9 + 7;
13760 ll max(ll a, ll b) { return ((a > b) ? a : b); }

```

```

13761 ll min(ll a, ll b) { return ((a > b) ? b : a); }
13762 mt19937 rng(chrono::steady_clock::now().time_since_epoch().count());
13763 void sol(void)
13764 {
13765     int n, m;
13766     cin >> n >> m;
13767     vll v(m);
13768     for (int i = 0; i < m; i++)
13769         cin >> v[i];
13770     sort(all(v));
13771     bool ingraph[m];
13772     memset(ingraph, false, sizeof(ingraph));
13773     ll relax = 0, ans = 0, curr_nodes = 1,skippedtillnow=0;
13774     for (int i = 0; i < m; i++)
13775     {
13776         if(skippedtillnow==1-curr_nodes+((curr_nodes)*(curr_nodes-1))/2)
13777         {
13778             ingraph[i]=true;
13779             curr_nodes++;
13780         }
13781         else
13782             skippedtillnow++;
13783     }
13784     for (int i = m-1; i >= 0; i--)
13785     {
13786         if (curr_nodes == n)
13787             break;
13788         if (!ingraph[i])
13789         {
13790             ingraph[i] = 1;
13791             curr_nodes++;
13792         }
13793     }
13794     for (int i = 0; i < m; i++)
13795     {
13796         if (ingraph[i])
13797             ans += v[i];
13798     }
13799     cout << ans << '\n';
13800 }
13801 int main()
13802 {
13803     ios_base::sync_with_stdio(false);
13804     cin.tie(NULL), cout.tie(NULL);
13805     int test = 1;
13806     cin >> test;
13807     while (test--)
13808         sol();
13809 }
13810
13811 //XORR
13812 //Utkarsh.25dec
13813 #include <bits/stdc++.h>
13814 #define ll long long int
13815 #define pb push_back
13816 #define mp make_pair
13817 #define mod 1000000007
13818 #define vl vector <ll>
13819 #define all(c) (c).begin(), (c).end()
13820 using namespace std;
13821 ll power(ll a,ll b) {ll res=1;a%=mod; assert(b>=0);
13822 for(;b;b>>=1){if(b&1)res=res*a%mod;a=a*a%mod;}return res;}
13823 ll modInverse(ll a){return power(a,mod-2);}
13824 const int N=500023;
13825 bool vis[N];
13826 vector <int> adj[N];
13827 long long readInt(long long l,long long r,char endd){
13828     long long x=0;
13829     int cnt=0;

```

```

13829     int fi=-1;
13830     bool is_neg=false;
13831     while(true){
13832         char g=getchar();
13833         if(g=='-'){
13834             assert(fi==-1);
13835             is_neg=true;
13836             continue;
13837         }
13838         if('0'<=g && g<='9'){
13839             x*=10;
13840             x+=g-'0';
13841             if(cnt==0){
13842                 fi=g-'0';
13843             }
13844             cnt++;
13845             assert(fi!=0 || cnt==1);
13846             assert(fi!=0 || is_neg==false);

13847
13848             assert(!(cnt>19 || ( cnt==19 && fi>1) ));
13849         } else if(g==endd){
13850             if(is_neg){
13851                 x= -x;
13852             }

13853
13854             if(!(l <= x && x <= r))
13855             {
13856                 cerr << l << ' ' << r << ' ' << x << '\n';
13857                 assert(l == 0);
13858             }

13859
13860             return x;
13861         } else {
13862             assert(false);
13863         }
13864     }
13865 }
13866 string readString(int l,int r,char endd){
13867     string ret="";
13868     int cnt=0;
13869     while(true){
13870         char g=getchar();
13871         assert(g!=-1);
13872         if(g==endd){
13873             break;
13874         }
13875         cnt++;
13876         ret+=g;
13877     }
13878     assert(l<=cnt && cnt<=r);
13879     return ret;
13880 }
13881 long long readIntSp(long long l,long long r){
13882     return readInt(l,r,' ');
13883 }
13884 long long readIntLn(long long l,long long r){
13885     return readInt(l,r,'\n');
13886 }
13887 string readStringLn(int l,int r){
13888     return readString(l,r,'\n');
13889 }
13890 string readStringSp(int l,int r){
13891     return readString(l,r,' ');
13892 }
13893 int maxAi=((1<<30)-1);
13894 int sumN=0;
13895 int A[N],B[N],C[N];
13896 void solve()
13897 {

```

```

13898     int n=readInt(1,300000,'\n');
13899     sumN+=n;
13900     assert(sumN<=300000);
13901     for(int i=1;i<=n;i++)
13902     {
13903         if(i==n)
13904             A[i]=readInt(0,maxAi,'\n');
13905         else
13906             A[i]=readInt(0,maxAi,' ');
13907     }
13908     for(int i=1;i<=n;i++)
13909     {
13910         if(i==n)
13911             B[i]=readInt(0,maxAi,'\n');
13912         else
13913             B[i]=readInt(0,maxAi,' ');
13914     }
13915     set<int> s;
13916     map<int,ll> cnt;
13917     for(int i=1;i<=n;i++)
13918     {
13919         C[i]=(A[i]^B[i]);
13920         s.insert(C[i]);
13921         cnt[C[i]]++;
13922     }
13923     ll ans=0;
13924     for(auto it:s)
13925         ans+=((cnt[it]*(cnt[it]-1))/2);
13926     cout<<ans<<'\n';
13927 }
13928 int main()
13929 {
13930     #ifndef ONLINE_JUDGE
13931     freopen("input.txt", "r", stdin);
13932     freopen("output.txt", "w", stdout);
13933     #endif
13934     ios_base::sync_with_stdio(false);
13935     cin.tie(NULL),cout.tie(NULL);
13936     int T=readInt(1,100000,'\n');
13937     while(T--)
13938         solve();
13939     assert(getchar()==-1);
13940     cerr << "Time : " << 1000 * ((double)clock()) / (double)CLOCKS_PER_SEC << "ms\n";
13941 }
13942
13943 // #include "bits/stdc++.h"
13944 using namespace std;
13945 #define ll long long
13946 #define pb push_back
13947 #define all(_obj) _obj.begin(), _obj.end()
13948 #define F first
13949 #define S second
13950 #define pll pair<ll, ll>
13951 #define vll vector<ll>
13952 ll INF = 1e18;
13953 const int N = 1e5 + 11, mod = 1e9 + 7;
13954 ll max(ll a, ll b) { return ((a > b) ? a : b); }
13955 ll min(ll a, ll b) { return ((a > b) ? b : a); }
13956 mt1937 rng(chrono::steady_clock::now().time_since_epoch().count());
13957 void sol(void)
13958 {
13959     ll n, ans = 0;
13960     cin >> n;
13961     map<ll, ll> cnt;
13962     vll a(n), b(n);
13963     for (int i = 0; i < n; i++)
13964     {
13965         cin >> a[i];
13966     }

```

```

13967     for (int i = 0; i < n; i++)
13968     {
13969         cin >> b[i];
13970         ans += cnt[a[i] ^ b[i]];
13971         cnt[b[i] ^ a[i]]++;
13972     }
13973     cout << ans << '\n';
13974     return;
13975 }
13976 int main()
13977 {
13978     ios_base::sync_with_stdio(false);
13979     cin.tie(NULL), cout.tie(NULL);
13980     int test = 1;
13981     cin >> test;
13982     while (test--)
13983         sol();
13984 }
13985
13986 //LOVE7
13987
13988 // Code by Sahil Tiwari (still_me)
13989
13990 #include <bits/stdc++.h>
13991 #define endl "\n"
13992 #define int long long int
13993 #define tt \
13994     int TESTCASE; \
13995     cin >> TESTCASE; \
13996     while (TESTCASE--)
13997 #define arrin(a, n) \
13998     for (int INPUT = 0; INPUT < n; INPUT++) \
13999     cin >> a[INPUT]
14000
14001 using namespace std;
14002 const int mod = 1e9 + 7;
14003
14004 const int N = 1000001;
14005
14006 // array to store inverse of 1 to N
14007 int factorialNumInverse[N + 1];
14008
14009 // array to precompute inverse of 1! to N!
14010 int naturalNumInverse[N + 1];
14011
14012 // array to store factorial of first N numbers
14013 int fact[N + 1];
14014
14015 // Function to precompute inverse of numbers
14016 void InverseofNumber(int p)
14017 {
14018     naturalNumInverse[0] = naturalNumInverse[1] = 1;
14019     for (int i = 2; i <= N; i++)
14020         naturalNumInverse[i] = naturalNumInverse[p % i] * (p - p / i) % p;
14021 }
14022 // Function to precompute inverse of factorials
14023 void InverseofFactorial(int p)
14024 {
14025     factorialNumInverse[0] = factorialNumInverse[1] = 1;
14026
14027     // precompute inverse of natural numbers
14028     for (int i = 2; i <= N; i++)
14029         factorialNumInverse[i] = (naturalNumInverse[i] * factorialNumInverse[i - 1]) % p;
14030 }
14031
14032 // Function to calculate factorial of 1 to N
14033 void factorial(int p)
14034 {
14035     fact[0] = 1;

```

```

14036
14037     // precompute factorials
14038     for (int i = 1; i <= N; i++)
14039     {
14040         fact[i] = (fact[i - 1] * i) % p;
14041     }
14042 }
14043
14044 // Function to return nCr % p in O(1) time
14045 int Binomial(int N, int R, int p)
14046 {
14047     // n C r = n!*inverse(r!)*inverse((n-r)!)
14048     if (N < R)
14049         return 0;
14050     if (N == R)
14051         return 1;
14052     int ans = ((fact[N] * factorialNumInverse[R]) % p * factorialNumInverse[N - R]) % p;
14053     return ans;
14054 }
14055
14056 bool ifsum(vector<int> a, int j, int k, map<pair<int, int>, int> &b)
14057 {
14058     if (k == 0)
14059         return 1;
14060     if (k < 0)
14061         return 0;
14062     if (j == a.size())
14063         return 0;
14064     if (b.find({j, k}) != b.end())
14065         return b[{j, k}];
14066     return b[{j, k}] = ifsum(a, j + 1, k - a[j], b) || ifsum(a, j + 1, k, b);
14067 }
14068
14069 void create(vector<int> a, vector<vector<int>> &b, int k)
14070 {
14071     if (k == 0)
14072     {
14073         map<pair<int, int>, int> c;
14074         if (!(ifsum(a, 0, 7, c) || ifsum(a, 0, 14, c) || ifsum(a, 0, 21, c) || ifsum(a,
14075             0, 28, c) || ifsum(a, 0, 35, c) || ifsum(a, 0, 42, c)))
14076             b.push_back(a);
14077         return;
14078     }
14079     for (int i = 1; i < 6; i++)
14080     {
14081         vector<int> c = a;
14082         c.push_back(i);
14083         create(c, b, k - 1);
14084     }
14085 }
14086
14087 int calc(vector<int> y, vector<int> x)
14088 {
14089     int ans = 1;
14090     for (int i = 0; i < 7; i++)
14091     {
14092         ans = (ans * Binomial(x[i], y[i], mod)) % mod;
14093     }
14094     return ans;
14095 }
14096
14097 vector<int> change(vector<int> a)
14098 {
14099     vector<int> y(7);
14100     for (int i : a)
14101         y[i]++;
14102     return y;
14103 }
14104
14105 signed main()

```

```

14104 {
14105     ios_base::sync_with_stdio(false);
14106     cin.tie(NULL);
14107     cout.tie(NULL);
14108
14109     int p = 1000000007;
14110     InverseofNumber(p);
14111     InverseofFactorial(p);
14112     factorial(p);
14113
14114     vector<vector<int>> a;
14115     for (int i = 1; i <= 5; i++)
14116     {
14117         vector<int> b;
14118         create(b, a, i);
14119     }
14120     set<vector<int>> c;
14121     for (vector<int> i : a)
14122     {
14123         vector<int> x = change(i);
14124         c.insert(x);
14125     }
14126
14127     tt
14128     {
14129         int n , q;
14130         cin >> n >> q;
14131         vector<int> x(7);
14132         int temp;
14133         for (int i = 0; i < n; i++)
14134         {
14135             cin >> temp;
14136             x[temp % 7]++;
14137         }
14138         while (q--)
14139         {
14140             int k;
14141             cin >> k;
14142             int ans = 0;
14143             if (k >= 6)
14144             {
14145                 for (int i = 1; i < 7; i++)
14146                     ans = (ans + Binomial(x[i], k, mod)) % mod;
14147             }
14148             else
14149             {
14150                 for (vector<int> i : c)
14151                 {
14152                     int t = accumulate(i.begin(), i.end(), 0LL);
14153                     if (t == k)
14154                     {
14155                         int xx = calc(i, x);
14156                         ans = (ans + xx) % mod;
14157                     }
14158                 }
14159             }
14160             cout << ans << endl;
14161         }
14162     }
14163     cerr << "time taken : " << (float)clock() / CLOCKS_PER_SEC << " secs" << endl;
14164     return 0;
14165 }
14166
14167 //LOVE7-EDITOR
14168
14169 #include "bits/stdc++.h"
14170 using namespace std;
14171 #define ll long long
14172 #define pb push_back

```



```

14173 #define all(_obj) _obj.begin(), _obj.end()
14174 #define F first
14175 #define S second
14176 #define pll pair<ll, ll>
14177 #define vll vector<ll>
14178 ll INF = 1e18;
14179 const int N = 1e5 + 11, mod = 1e9 + 7;
14180 ll max(ll a, ll b) { return ((a > b) ? a : b); }
14181 ll min(ll a, ll b) { return ((a > b) ? b : a); }
14182 mt19937 rng(chrono::steady_clock::now().time_since_epoch().count());
14183 ll factorial[N], inverse_factorial[N], NumInverse[N];
14184 set<vector<int>>> all;
14185 ll binomial_coefficient(int n, int k)
14186 {
14187     if (n < k)
14188         return 0;
14189     return factorial[n] * inverse_factorial[k] % mod * inverse_factorial[n - k] % mod;
14190 }
14191 void add(vector<int> &v)
14192 {
14193     for (int i = v.size() - 1; i >= 0; i--)
14194         if (v[i] < 6)
14195         {
14196             v[i]++;
14197             return;
14198         }
14199     else
14200         v[i] = 1;
14201     return;
14202 }
14203 void sol(void)
14204 {
14205     int n, q;
14206     cin >> n >> q;
14207     vll a(n);
14208     int cnt[7];
14209     memset(cnt, 0, sizeof(cnt));
14210     for (int i = 0; i < n; i++)
14211     {
14212         cin >> a[i];
14213         a[i] %= 7;
14214         cnt[a[i]]++;
14215     }
14216     ll ansfor[8];
14217     memset(ansfor, 0, sizeof(ansfor));
14218     for (int k = 1; k <= 7; k++)
14219     {
14220         for (auto x : all)
14221         {
14222             if (x.size() != k)
14223                 continue;
14224             ll calc = 1;
14225             int temp[7];
14226             memset(temp, 0, sizeof(temp));
14227             for (auto y : x)
14228                 temp[y]++;
14229             for (int i = 1; i <= 6; i++)
14230                 calc *= binomial_coefficient(cnt[i], temp[i]), calc %= mod;
14231             ansfor[k] += calc;
14232             ansfor[k] %= mod;
14233         }
14234     }
14235     while (q--)
14236     {
14237         int k;
14238         cin >> k;
14239         ll ans = 0, calc;
14240         if (k > 7)
14241             {

```

```

14242         for (int i = 1; i <= 6; i++)
14243             ans += binomial_coefficient(cnt[i], k);
14244         ans %= mod;
14245     }
14246     else
14247     {
14248         ans = ansfor[k];
14249     }
14250     cout << ans << '\n';
14251 }
14252 return;
14253 }
14254 int main()
14255 {
14256     ios_base::sync_with_stdio(false);
14257     cin.tie(NULL), cout.tie(NULL);
14258     NumInverse[0] = NumInverse[1] = 1;
14259     factorial[0] = factorial[1] = 1;
14260     inverse_factorial[0] = inverse_factorial[1] = 1;
14261     for (int i = 2; i < N; i++)
14262     {
14263         NumInverse[i] = NumInverse[mod % i] * (mod - mod / i) % mod;
14264         factorial[i] = factorial[i - 1] * i % mod;
14265         inverse_factorial[i] = (NumInverse[i] * inverse_factorial[i - 1]) % mod;
14266     }
14267     for (int sz = 1; sz <= 7; sz++)
14268     {
14269         vector<int> v(sz, 1), finalv(sz, 6);
14270         while (true)
14271         {
14272             bool flag = true;
14273             for (int i = 0; i < (1 << sz); i++)
14274             {
14275                 int sum = 0, bitcount = 0;
14276                 for (int j = 0; j < sz; j++)
14277                     if (i & (1 << j))
14278                         sum += v[j], bitcount++;
14279                 if ((sum % 7 == 0 && bitcount % 7 != 0) || (sum % 7 != 0 && bitcount % 7 == 0))
14280                 {
14281                     flag = false;
14282                     break;
14283                 }
14284             }
14285             if (flag)
14286             {
14287                 vector<int> vv = v;
14288                 sort(all(vv));
14289                 all.insert(vv);
14290             }
14291             if (v == finalv)
14292                 break;
14293             add(v);
14294         }
14295     }
14296     int test = 1;
14297     cin >> test;
14298     while (test--)
14299         sol();
14300 }
14301
14302 //ORPERM
14303 /**
14304  *
14305  * Author    : Sai Suman Chitturi
14306  * Handle    : suman_18733097
14307  *
14308  **/
14309

```

```

14310 #include <bits/stdc++.h>
14311 using namespace std;
14312
14313 void solve() {
14314     int N = 0;
14315     cin >> N;
14316
14317     vector<int> a(N), b(N);
14318
14319     for(int i = 0; i < N; i++) {
14320         cin >> a[i];
14321     }
14322     for(int i = 0; i < N; i++) {
14323         cin >> b[i];
14324     }
14325
14326     int X = b[0];
14327     for(int i = 0; i < N; i++) {
14328         X &= b[i];
14329     }
14330
14331     for(int i = 0; i < N; i++) {
14332         a[i] |= X;
14333     }
14334
14335     sort(a.begin(), a.end());
14336     sort(b.begin(), b.end());
14337
14338     for(int i = 0; i < N; i++) {
14339         if(a[i] != b[i]) {
14340             cout << -1 << '\n';
14341             return;
14342         }
14343     }
14344
14345     cout << X << '\n';
14346 }
14347
14348 int main() {
14349     ios_base::sync_with_stdio(false);
14350     cin.tie(NULL); cout.tie(NULL);
14351
14352     int t = 1;
14353     cin >> t;
14354
14355     for(int test = 0; test < t; test++) {
14356         solve();
14357     }
14358
14359     return 0;
14360 }
14361
14362 //ORPERM-EDITOR
14363 #include "bits/stdc++.h"
14364 using namespace std;
14365 #define ll long long
14366 #define pb push_back
14367 #define all(_obj) _obj.begin(), _obj.end()
14368 #define F first
14369 #define S second
14370 #define pll pair<ll, ll>
14371 #define vll vector<ll>
14372 ll INF = 1e18;
14373 const int N = 1e5 + 11, mod = 1e9 + 7;
14374 ll max(ll a, ll b) { return ((a > b) ? a : b); }
14375 ll min(ll a, ll b) { return ((a > b) ? b : a); }
14376 mt19937 rng(chrono::steady_clock::now().time_since_epoch().count());
14377 void sol(void)
14378 {

```

```

14379     int n;
14380     cin >> n;
14381     vll a(n), b(n);
14382     int bita[30], bitb[30];
14383     memset(bita, 0, sizeof(bita));
14384     memset(bitb, 0, sizeof(bitb));
14385     for (int i = 0; i < n; i++)
14386     {
14387         cin >> a[i];
14388         for (int j = 0; j < 30; j++)
14389             if (a[i] & (1 << j))
14390                 bita[j]++;
14391     }
14392     for (int i = 0; i < n; i++)
14393     {
14394         cin >> b[i];
14395         for (int j = 0; j < 30; j++)
14396             if (b[i] & (1 << j))
14397                 bitb[j]++;
14398     }
14399     int x = 0;
14400     for (int i = 0; i < 30; i++)
14401     {
14402         if (bitb[i] < bita[i] || (bita[i] < bitb[i] && bitb[i] != n))
14403         {
14404             x = -1;
14405             break;
14406         }
14407         if (bitb[i] == bita[i])
14408             continue;
14409         x += (1 << i);
14410     }
14411     sort(all(b));
14412     for (int i = 0; i < n; i++)
14413         a[i] |= x;
14414     sort(all(a));
14415     if (a == b)
14416         cout << x << '\n';
14417     else
14418         cout << -1 << '\n';
14419 }
14420 int main()
14421 {
14422     ios_base::sync_with_stdio(false);
14423     cin.tie(NULL), cout.tie(NULL);
14424     int test = 1;
14425     cin >> test;
14426     while (test--)
14427         sol();
14428 }
14429
14430 //DOUBTSUPPORT
14431 #include<bits/stdc++.h>
14432 using namespace std;
14433 typedef long long ll;
14434 #define fi first
14435 #define se second
14436 const int iu=10000;
14437 const ll mod=998244353;
14438 ll n;
14439 int a[2001];
14440 // ----- Input Checker Start -----
14441
14442 long long readInt(long long l, long long r, char endd)
14443 {
14444     long long x = 0;
14445     int cnt = 0, fi = -1;
14446     bool is_neg = false;
14447     while(true)

```

```

14448     {
14449         char g = getchar();
14450         if(g == '-')
14451         {
14452             assert(fi == -1);
14453             is_neg = true;
14454             continue;
14455         }
14456         if('0' <= g && g <= '9')
14457         {
14458             x *= 10;
14459             x += g - '0';
14460             if(cnt == 0)
14461                 fi = g - '0';
14462             cnt++;
14463             assert(fi != 0 || cnt == 1);
14464             assert(fi != 0 || is_neg == false);
14465             assert(!(cnt > 19 || (cnt == 19 && fi > 1)));
14466         }
14467         else if(g == endd)
14468         {
14469             if(is_neg)
14470                 x = -x;
14471             if(!(l <= x && x <= r))
14472             {
14473                 cerr << l << ' ' << r << ' ' << x << '\n';
14474                 assert(false);
14475             }
14476             return x;
14477         }
14478         else
14479         {
14480             assert(false);
14481         }
14482     }
14483 }
14484
14485 string readString(int l, int r, char endd)
14486 {
14487     string ret = "";
14488     int cnt = 0;
14489     while(true)
14490     {
14491         char g = getchar();
14492         assert(g != -1);
14493         if(g == endd)
14494             break;
14495         cnt++;
14496         ret += g;
14497     }
14498     assert(l <= cnt && cnt <= r);
14499     return ret;
14500 }
14501
14502 long long readIntSp(long long l, long long r) { return readInt(l, r, ' '); }
14503 long long readIntLn(long long l, long long r) { return readInt(l, r, '\n'); }
14504 string readStringLn(int l, int r) { return readString(l, r, '\n'); }
14505 string readStringSp(int l, int r) { return readString(l, r, ' '); }
14506 void readEOF() { assert(getchar() == EOF); }
14507
14508 vector<int> readVectorInt(int n, long long l, long long r)
14509 {
14510     vector<int> a(n);
14511     for(int i = 0; i < n - 1; i++)
14512         a[i] = readIntSp(l, r);
14513     a[n - 1] = readIntLn(l, r);
14514     return a;
14515 }
14516

```

```

14517 // ----- Input Checker End -----
14518 void solve() {
14519     ll x,y,z;
14520     x=readInt(1,100,' ');y=readInt(1,100,' ');z=readInt(1,100,'\n');
14521     cout << (z>=x)+(z>=x+y) << '\n';
14522 }
14523 int main() {
14524     ios::sync_with_stdio(false);
14525     ll a = readIntLn(1, 5000);
14526     readEOF();
14527     if(a<=1600)
14528         cout<<"yeS\n";
14529     else
14530         cout<<"nO\n";
14531 }
14532
14533 //DOUBTSUPPORT-EDITOR
14534 #include "bits/stdc++.h"
14535 using namespace std;
14536 #define ll long long
14537 #define pb push_back
14538 #define all(_obj) _obj.begin(), _obj.end()
14539 #define F first
14540 #define S second
14541 #define pll pair<ll, ll>
14542 #define vll vector<ll>
14543 ll INF = 1e18;
14544 const int N = 1e5 + 11, mod = 1e9 + 7;
14545 ll max(ll a, ll b) { return ((a > b) ? a : b); }
14546 ll min(ll a, ll b) { return ((a > b) ? b : a); }
14547 mt19937 rng(chrono::steady_clock::now().time_since_epoch().count());
14548 void sol(void)
14549 {
14550     int D;
14551     cin >> D;
14552     cout << ((D <= 1600) ? "YES" : "NO");
14553     return;
14554 }
14555 int main()
14556 {
14557     ios_base::sync_with_stdio(false);
14558     cin.tie(NULL), cout.tie(NULL);
14559     int test = 1;
14560     // cin>>test;
14561     while (test--)
14562         sol();
14563 }
14564
14565 //DISCOUNTT
14566 //Utkarsh.25dec
14567 #include <bits/stdc++.h>
14568 #define ll long long int
14569 #define pb push_back
14570 #define mp make_pair
14571 #define mod 1000000007
14572 #define vl vector<ll>
14573 #define all(c) (c).begin(), (c).end()
14574 using namespace std;
14575 ll power(ll a,ll b) {ll res=1;a%=mod; assert(b>=0);
14576     for(;b;b>>=1){if(b&1) res=res*a%mod;a=a*a%mod;}return res;}
14577 ll modInverse(ll a){return power(a,mod-2);}
14578 const int N=500023;
14579 bool vis[N];
14580 vector<int> adj[N];
14581 long long readInt(long long l,long long r,char endd){
14582     long long x=0;
14583     int cnt=0;
14584     int fi=-1;
14585     bool is_neg=false;

```

```

14585     while(true){
14586         char g=getchar();
14587         if(g=='-'){
14588             assert(fi==-1);
14589             is_neg=true;
14590             continue;
14591         }
14592         if('0'<=g && g<='9'){
14593             x*=10;
14594             x+=g-'0';
14595             if(cnt==0){
14596                 fi=g-'0';
14597             }
14598             cnt++;
14599             assert(fi!=0 || cnt==1);
14600             assert(fi!=0 || is_neg==false);
14601
14602             assert(!(cnt>19 || ( cnt==19 && fi>1) ));
14603         } else if(g==endd){
14604             if(is_neg){
14605                 x= -x;
14606             }
14607
14608             if(!(l <= x && x <= r))
14609             {
14610                 cerr << l << ' ' << r << ' ' << x << '\n';
14611                 assert(1 == 0);
14612             }
14613
14614             return x;
14615         } else {
14616             assert(false);
14617         }
14618     }
14619 }
14620 string readString(int l,int r,char endd){
14621     string ret="";
14622     int cnt=0;
14623     while(true){
14624         char g=getchar();
14625         assert(g!=-1);
14626         if(g==endd){
14627             break;
14628         }
14629         cnt++;
14630         ret+=g;
14631     }
14632     assert(l<=cnt && cnt<=r);
14633     return ret;
14634 }
14635 long long readIntSp(long long l,long long r){
14636     return readInt(l,r,' ');
14637 }
14638 long long readIntLn(long long l,long long r){
14639     return readInt(l,r,'\n');
14640 }
14641 string readStringLn(int l,int r){
14642     return readString(l,r,'\n');
14643 }
14644 string readStringSp(int l,int r){
14645     return readString(l,r,' ');
14646 }
14647 void solve()
14648 {
14649     int n=readInt(1,100,' ');
14650     int X=readInt(1,100000,' ');
14651     int Y=readInt(1,100000,'\n');
14652     int A[n+1]={0};
14653     int withoutdisc=0,withdisc=X;

```

```

14654     for(int i=1;i<=n;i++)
14655     {
14656         if(i==n)
14657             A[i]=readInt(1,100000,'\n');
14658         else
14659             A[i]=readInt(1,100000,' ');
14660         withoutdisc+=A[i];
14661         withdisc+=(max(A[i]-Y,0));
14662     }
14663     if(withdisc<withoutdisc)
14664         cout<<"COUPON\n";
14665     else
14666         cout<<"NO COUPON\n";
14667 }
14668
14669 //DISCOUNTT-EDITOR
14670 #include "bits/stdc++.h"
14671 using namespace std;
14672 #define ll long long
14673 #define pb push_back
14674 #define all(_obj) _obj.begin(), _obj.end()
14675 #define F first
14676 #define S second
14677 #define pll pair<ll, ll>
14678 #define vll vector<ll>
14679 ll INF = 1e18;
14680 const int N = 1e5 + 11, mod = 1e9 + 7;
14681 ll max(ll a, ll b) { return ((a > b) ? a : b); }
14682 ll min(ll a, ll b) { return ((a > b) ? b : a); }
14683 mt19937 rng(chrono::steady_clock::now().time_since_epoch().count());
14684 void sol(void)
14685 {
14686     int n, sum1 = 0, sum2 = 0, x, y;
14687     cin >> n >> x >> y;
14688     vll v(n);
14689     for (int i = 0; i < n; i++)
14690     {
14691         cin >> v[i], sum1 += v[i];
14692         sum2 += max(0, v[i] - y);
14693     }
14694     if (sum1 > sum2 + x)
14695         cout << "COUPON\n";
14696     else
14697         cout << "NO COUPON\n";
14698     return;
14699 }
14700 int main()
14701 {
14702     ios_base::sync_with_stdio(false);
14703     cin.tie(NULL), cout.tie(NULL);
14704     int test = 1;
14705     cin >> test;
14706     while (test--)
14707         sol();
14708 }
14709
14710 //KAJY
14711 #include "bits/stdc++.h"
14712 #define MAXN 100009
14713 #define INF 1000000007
14714 #define mp(x,y) make_pair(x,y)
14715 #define all(v) v.begin(),v.end()
14716 #define pb(x) push_back(x)
14717 #define wr cout<<"-----"<<endl;
14718 #define ppb() pop_back()
14719 #define tr(ii,c) for(__typeof((c).begin()) ii=(c).begin();ii!=(c).end();ii++)
14720 #define ff first
14721 #define ss second
14722 #define my_little_dodge 46

```



```

14723 #define debug(x)  cerr<< #x <<" = "<< x<<endl;
14724 using namespace std;
14725
14726 typedef long long ll;
14727 typedef pair<int,int> PII;
14728 template<class T>bool umin(T& a,T b){if(a>b){a=b;return 1;}return 0;}
14729 template<class T>bool umax(T& a,T b){if(a<b){a=b;return 1;}return 0;}
14730 int a[MAXN],b[MAXN],c[MAXN];
14731 int n,q;
14732 ll A[MAXN],B[MAXN],C[MAXN];
14733 struct node{
14734     ll ans;
14735     ll max_d,max_e;
14736 }st[MAXN<<2];
14737 ll lazy_e[MAXN<<2],lazy_d[MAXN<<2];
14738 node merge(node x,node y){
14739     node z;
14740     z.ans=max(x.ans,y.ans);
14741     umax(z.ans,x.max_d+y.max_e);
14742     z.max_d=max(x.max_d,y.max_d);
14743     z.max_e=max(x.max_e,y.max_e);
14744     return z;
14745 }
14746 void shift(int nd){
14747     ll &ret=lazy_d[nd],&pet=lazy_e[nd];
14748     st[nd<<1].ans+=ret+pet;
14749     st[nd<<1|1].ans+=ret+pet;
14750
14751     st[nd<<1].max_d+=ret;
14752     lazy_d[nd<<1]+=ret;
14753
14754     st[nd<<1|1].max_d+=ret;
14755     lazy_d[nd<<1|1]+=ret;
14756
14757     st[nd<<1].max_e+=pet;
14758     lazy_e[nd<<1]+=pet;
14759
14760     st[nd<<1|1].max_e+=pet;
14761     lazy_e[nd<<1|1]+=pet;
14762
14763     ret=0;pet=0;
14764 }
14765 void inc(int l,int r,int t,int val,int nd=1,int x=0,int y=n){
14766     if(l>y or x>r or l>r)return;
14767     if(l<=x and y<=r){
14768         if(!t)st[nd].max_d+=val,lazy_d[nd]+=val;
14769         else st[nd].max_e+=val,lazy_e[nd]+=val;
14770         st[nd].ans+=val;
14771         return;
14772     }
14773     shift(nd);
14774     int mid=(x+y)>>1;
14775     inc(l,r,t,val,nd<<1,x,mid);
14776     inc(l,r,t,val,nd<<1|1,mid+1,y);
14777     st[nd]=merge(st[nd<<1],st[nd<<1|1]);
14778 }
14779 void build(int nd=1,int x=0,int y=n){
14780     if(x==y){
14781         st[nd].ans=0;
14782         st[nd].max_d=A[x]-B[x];
14783         st[nd].max_e=C[x+1]+B[x];
14784         return;
14785     }
14786     int mid=(x+y)>>1;
14787     build(nd<<1,x,mid);
14788     build(nd<<1|1,mid+1,y);
14789     st[nd]=merge(st[nd<<1],st[nd<<1|1]);
14790 }
14791 int main(){

```

```

14792     //~ freopen("file.in","r",stdin);
14793     scanf("%d%d",&n,&q);
14794     for(int i=1;i<=n;i++)scanf("%d",a+i),A[i]=A[i-1]+a[i];
14795     for(int i=1;i<=n;i++)scanf("%d",b+i),B[i]=B[i-1]+b[i];
14796     for(int i=1;i<=n;i++)scanf("%d",c+i);
14797     for(int i=n;i>=1;i--)C[i]=C[i+1]+c[i];
14798     build();
14799     while(q--){
14800         int t,pos,val;
14801         scanf("%d%d%d",&t,&pos,&val);
14802         if(t==1)inc(pos,n,0,val-a[pos]),a[pos]=val;
14803         if(t==2)inc(pos,n,0,b[pos]-val),inc(pos,n,1,val-b[pos]),b[pos]=val;
14804         if(t==3)inc(1,pos-1,1,val-c[pos]),c[pos]=val;
14805         printf("%lld\n",st[1].ans);
14806     }
14807     return 0;
14808 }
14809
14810 //KAJY-EDITOR
14811
14812 #include "bits/stdc++.h"
14813 using namespace std;
14814 #define ll long long
14815 #define pb push_back
14816 #define all(_obj) _obj.begin(), _obj.end()
14817 #define F first
14818 #define S second
14819 #define pll pair<ll, ll>
14820 #define vll vector<ll>
14821 ll INF = 1e18;
14822 const int N = 1e5 + 11, mod = 1e9 + 7;
14823 ll max(ll a, ll b) { return ((a > b) ? a : b); }
14824 ll min(ll a, ll b) { return ((a > b) ? b : a); }
14825 mt19937 rng(chrono::steady_clock::now().time_since_epoch().count());
14826 using namespace std;
14827 int a[N], b[N], c[N];
14828 ll P[N], M[N], S[N];
14829 struct node
14830 {
14831     ll ans, max_d, max_e;
14832 } st[N << 2];
14833 ll lazy_e[N << 2], lazy_d[N << 2];
14834 node combine(node left, node right)
14835 {
14836     node root;
14837     root.ans = max(left.ans, right.ans);
14838     root.ans = max(left.max_d + right.max_e, root.ans);
14839     root.max_d = max(left.max_d, right.max_d);
14840     root.max_e = max(left.max_e, right.max_e);
14841     return root;
14842 }
14843 void shift(int nd)
14844 {
14845     ll &x = lazy_d[nd], &y = lazy_e[nd];
14846     st[nd << 1].ans += x + y;
14847     st[nd << 1 | 1].ans += x + y;
14848
14849     st[nd << 1].max_d += x;
14850     lazy_d[nd << 1] += x;
14851
14852     st[nd << 1 | 1].max_d += x;
14853     lazy_d[nd << 1 | 1] += x;
14854
14855     st[nd << 1].max_e += y;
14856     lazy_e[nd << 1] += y;
14857
14858     st[nd << 1 | 1].max_e += y;
14859     lazy_e[nd << 1 | 1] += y;
14860

```

```

14861     x = 0;
14862     y = 0;
14863 }
14864 void update(int l, int r, int t, int val, int nd, int x, int y)
14865 {
14866     if (l > y or x > r or l > r)
14867         return;
14868     if (l <= x and y <= r)
14869     {
14870         if (!t)
14871             st[nd].max_d += val, lazy_d[nd] += val;
14872         else
14873             st[nd].max_e += val, lazy_e[nd] += val;
14874         st[nd].ans += val;
14875         return;
14876     }
14877     shift(nd);
14878     int mid = (x + y) >> 1;
14879     update(l, r, t, val, nd << 1, x, mid);
14880     update(l, r, t, val, nd << 1 | 1, mid + 1, y);
14881     st[nd] = combine(st[nd << 1], st[nd << 1 | 1]);
14882 }
14883 void build(int nd, int x, int y)
14884 {
14885     if (x == y)
14886     {
14887         st[nd].ans = 0;
14888         st[nd].max_d = P[x] - M[x]; // D array
14889         st[nd].max_e = S[x + 1] + M[x]; // E array
14890         return;
14891     }
14892     int mid = (x + y) >> 1;
14893     build(nd << 1, x, mid);
14894     build(nd << 1 | 1, mid + 1, y);
14895     st[nd] = combine(st[nd << 1], st[nd << 1 | 1]);
14896 }
14897 void sol(void)
14898 {
14899     int n, q;
14900     cin >> n >> q;
14901     for (int i = 1; i <= n; i++)
14902         cin >> a[i], P[i] = P[i - 1] + a[i];
14903     for (int i = 1; i <= n; i++)
14904         cin >> b[i], M[i] = M[i - 1] + b[i];
14905     for (int i = 1; i <= n; i++)
14906         cin >> c[i];
14907     for (int i = n; i >= 1; i--)
14908         S[i] = S[i + 1] + c[i];
14909     build(1, 0, n);
14910     while (q--)
14911     {
14912         int t, pos, val;
14913         cin >> t >> pos >> val;
14914         // 0 for array D and 1 for array E
14915         if (t == 1)
14916             update(pos, n, 0, val - a[pos], 1, 0, n), a[pos] = val;
14917         if (t == 2)
14918             update(pos, n, 0, b[pos] - val, 1, 0, n), update(pos, n, 1, val - b[pos],
14919             1, 0, n), b[pos] = val;
14920         if (t == 3)
14921             update(1, pos - 1, 1, val - c[pos], 1, 0, n), c[pos] = val;
14922         cout << st[1].ans << '\n';
14923     }
14924     return;
14925 }
14926 int main()
14927 {
14928     ios_base::sync_with_stdio(false);

```

```
14929         cin.tie(NULL), cout.tie(NULL);
14930         int test = 1;
14931         // cin>>test;
14932         while (test--)
14933             sol();
14934     }
14935
14936     //TILL START40
14937     //TO START START39
14938
14939
14940
14941
14942
14943
14944
14945
14946
```