SOLVE LATER

Battlefield (MEDIUM)

Attempted by: **277** / Accuracy: **35%** / Maximum Score: **30** /
10 Votes

Tag(s): Easy-Medium

Our main task for this problem is to bring all K's and D's together in minimum number of swaps.For this we can count number of D's or number of K's in given input and then can construct window of that given length ,and for then all window can check how many D's or K's are missing to get the minimum number of swaps.

```
//--------------*************------------
---------
/*
 #           "      ""#      ""#
#
 #    m   mmm         #        #       mmm
mmm      mmm#    mmm
 # m"       #        #        #     #"   "   #"
"#   #"  "#   #"   #
 #"#         #        #        #     #        #
#   #    #   #""""
 #   "m   mm#mm      "mm      "mm   "#mm"
"#m#"   "#m##   "#mm"   */


#include<bits/stdc++.h>
using namespace std;
#define pb push_back
#define mp make_pair
typedef pair<int,int> pii;
typedef long long ll;
typedef double ld;
typedef vector<int> vi;
#define fi first
#define se second
```

BEST SUBMISSIONS

LANGUAGE: C

TIME (sec)	MEMORY (KiB)
0.70454	64

by Christophe Savard

VIEW BEST SUBMISSION VIEW ALL SUBMISSION

CONTRIBUTOR

AUTHOR
Atul Mishra

SOCIAL SHARE

```
#define fe first
#define SZ 666666
#define si(n) scanf("%d",&n);
#define sl(n) scanf("%ld",&n);
#define pi(n) printf("%d\n",n);
#define pl(n) printf("%ld\n",n);
#define pf(n) printf("%f\n",n);
#define FILL(a,b) memset(a,0,sizeof(b));
#define rep(i,n) for(int i=0;i<n;i++)
#define reps(i,a,b) for(int i=1;i<=b;i++)
const int INF=1e9+5;
const int MOD=1000000007;

//--------------*************------------
---------

int main(){
  int T;
  scanf("%d",&T);
    while(T--){
         int N,k=0,countK=0;
         string slide;

         scanf("%d",&N);
         cin>>slide;

         slide+=slide;

         for(int
i=0;i<N;i++)if(slide[i]=='K')++k;

         for(int i=0;i<k;i++){
              if(slide[i]=='K')++countK;
         }
         int  maxK = countK;

         for(int i=k;i<(N+N);i++){
              countK = countK +
((slide[i]=='K')?1:0) -
((slide[i-k]=='K')?1:0);
              //printf("%d\n",countK);
              maxK = max(maxK,countK);
         }

        printf("%d\n",(k-maxK));
    }
 return 0;
 }
```