Submit:	`bit.c` / `bit.cpp` / `bit.pas`	Časový limit:	řádově sekundy
Vstup:	`bit.in`	Paměťový limit:	32 MB
Výstup:	`bit.out`	Bodů:	100

Věhlasní a veleváženi vědci vévodství Bajtlandského se rozhodli vyvinout nový typ počítače – počítač tříbitový (brzy dostal přezdívku Trojlobit, zkráceně TBC). Předpověděli, že tento počítač bude mít dostatečný výkon na vyřešení všech problémů, které lidstvo trápí od prvohor až po současnost. Zbývá jim nicméně vyřešit několik drobných technických detailů, a tak si vás přizvali na pomoc s jedním z nich.

Bajtlandští nyní pracují na proceduře pro inicializování paměti stroje. Ta sestává z n bitů očíslovaných od 1 do n a každý bit může nabývat jedné z hodnot a, b, c nebo být neinicializovaný. Počítač umí provést následující inicializační operace:

- dva po sobě jdoucí neinicializované bity, tj. bity s čísly i a $i+1$ ($1 \le i < n$) mohou být nastaveny na *rozdílné* hodnoty.
- dva po sobě jdoucí bity, z nichž jeden je inicializovaný na nějakou hodnotu x a druhý neinicializovaný, mohou být oba nastaveny na hodnoty různé od x.

Například je tedy možné inicializovat takto (zde je $n = 4$ a u značí neinicializovaný bit): $uuuu \to uuab \to ucbb \to babb$.

Úloha: Napište program, který:

- přečte hodnoty, na které má být paměť inicializována,
- zjistí, jestli je inicializace možná, a
- zapíše to do výstupu.

Vstup: Vstupní soubor `bit.in` obsahuje několik inicializačních vzorků. Na prvním řádku je jediné celé kladné číslo N ($1 \le N \le 10$), což je počet vzorků. Následují jednotlivé vzorky, každý z nich na dvou po sobě jdoucích řádcích – na prvním řádku je celé číslo l_i ($1 \le l_i \le 100\,000$), což je délka i-tého vzorku (čili velikost paměti počítače), na druhém pak řetězec l_i písmen a, b, c, což je vzorek sám, tedy posloupnost hodnot, na kterou se má paměť inicializovat.

Výstup: Výstupní soubor se jmenuje `bit.out`. Zapište do něj N řádků: i-tý řádek odpovídá i-tému vzorku a obsahuje buďto slovo "`YES`", pokud je inicializace možná, nebo "`NO`", pokud nikoliv.

Příklad:

bit.in	bit.out
2	NO
4	YES
aabb	
4	
aaab	