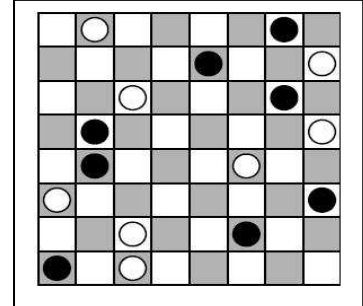


**XXI РЕПУБЛИКАНСКА СТУДЕНТСКА ОЛИМПИАДА ПО ПРОГРАМИРАНЕ**  
**Варна, 16.05.2009**

**Задача С. ИГРА**

В условията на финансова криза е много важно да си направиш добре сметката – дори когато играеш на простичка игра, като описаната по-долу. Двама играчи – Бял и Черен – играят игра на квадратна дъска с  $N$  реда и  $N$  стълба, като се редуват да правят ходове. Във всеки ред на дъската има по един бял и един черен пул. Белият играч има право да мести само белите пулове, а Черният – само черните. Пул може да се мести само по реда, в който се намира, в посока към пула на противника. Играчът, който е на ход, може да премести само един от своите пулове на всяко едно празно квадратче в реда, но няма право да застава върху другият пул или да го прескочи. Губи този играч, който е на ход и не може да премести пул. Напишете програма, която определя победителя в играта, ако се знае, че Белият играч е винаги първи на ход и двамата играчи играят оптимално.



**Вход.** На първия ред на стандартния вход ще бъде зададен броят на тестовите примери. Всеки тестов пример започва с ред, на който е зададен размерът  $N$  на игралната дъска ( $2 \leq N \leq 1000$ ). Следват  $N$  реда, описващи началната ситуация на дъската. На  $i$ -тия от тези редове са зададени две числа – стълба, в който се намира белият пул на  $i$ -тия ред от дъската. Номерацията на редовете и стълбовете започва от 1.

**Изход.** За всеки тестов пример, в който Белият играч може да спечели, програмата трябва да изведе на един ред на стандартния изход думата **WHITE**, последвана от едно число – номера на реда от дъската, в който той трябва да направи своя печеливш ход. Думата и числото трябва да са разделени с един интервал. Ако съществуват няколко реда, в които Белият играч може да направи печеливш ход, програмата трябва да изведе този с минимален номер. Ако играта се печели от Черния играч, независимо какво играе Белият, програмата трябва да изведе за този тестов пример думата **BLACK**.

**ПРИМЕР**

<b>Вход</b>	<b>Изход</b>
2	WHITE 1 BLACK
8	
2 7	
8 5	
3 7	
8 2	
6 2	
1 8	
3 6	
3 1	
4	
1 2	
4 3	
3 2	
2 1	