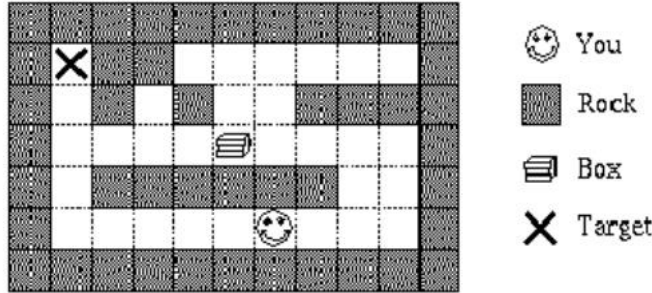


Đẩy hộp

Bạn hãy tưởng tượng bạn đang đứng trong mê cung 2 chiều gồm một ô có thể là đá hoặc không. Bạn có thể di chuyển theo các hướng lên trên, xuống dưới, sang trái hoặc sang phải tại mỗi bước. Mỗi bước di chuyển gọi là một bước đi.

Một trong các ô trống chứa một hộp. Chiếc hộp này có thể di chuyển sang ô kề cạnh bằng cách đứng ở phía đối diện và đẩy theo hướng di chuyển của hộp, mỗi bước di chuyển vậy gọi là một lần đẩy. Ngoài các lần đẩy, người chơi cần di chuyển không đẩy để thay đổi hướng đẩy của hộp. Vì hộp khá nặng nên bạn cần đẩy sao cho số lần đẩy là ít nhất.

Yêu cầu: Tìm số lần đẩy hộp ít nhất từ vị trí ban đầu đến đích và số bước di chuyển ít nhất (= tổng số bước di chuyển đẩy + số bước di chuyển không đẩy).



Dữ liệu: vào từ file pushing.inp gồm:

- Dòng đầu ghi kích thước M, N của ma trận ($1 < M.N \leq 21$).
- M dòng tiếp theo mỗi dòng ghi N kí tự của ma trận:
 - '.' thể hiện ô trống không đá
 - 'S' thể hiện ô đứng hiện tại
 - 'T' thể hiện ô đích
 - 'B' thể hiện ô chứa hộp
 - '#' thể hiện ô đá

Kết quả: ghi ra file pushing.out

- Impossible nếu không thể đẩy hộp đến đích
- Ngược lại ghi 2 số cách nhau bởi dấu cách: số lần đẩy ít nhất và số bước di chuyển ít nhất (= tổng số bước di chuyển đẩy + số bước di chuyển không đẩy) để đẩy hộp đến đích

Ví dụ:

Pushing.inp	Pushing.out
<pre> 7 11 ##### #T##.....# #.#.#.#### #....B....# ######..# #.....S...# ##### </pre>	<pre> 6 28 </pre>