

PROBLEME - LISTE

1. Într-o bibliotecă sunt mai multe teancuri de cărți. Bibliotecarul vrea să rețină doar cărțile având anul de apariție mai mare decât 1970, dar în ordine alfabetică a titlurilor. Să se afișeze cărțile în ordinea în care trebuie reținute pe raft. **Indicație:** Stivă, CoadăCuPriorități.
2. **Jocul Gâsca Roșie.** 2 jucători primesc inițial $\frac{n}{2}$ cărți de joc (fiecare carte poate avea culoarea roșie sau neagră). Jucătorii pun alternativ câte o carte pe masă (din vârful teancului lor de cărți), până se pune o carte roșie (caz în care teancul de pe masă va fi luat de către jucătorul care nu a pus cartea roșie și adăugat sub teancul său de cărți). Pierde jucătorul care nu mai are cărți. Simulați jocul. **Indicație:** Stivă, Coadă.
3. Fie un labirint (rețea dreptunghiulară) cu celule ocupate (X) și libere (*). Fie R un robot în acest labirint. Robotul se poate deplasa în 4 direcții: N, S, E, V.

*	*	X	*	*
X	*	*	*	X
X	*	R	*	*
X	*	X	X	*
*	*	*	*	*

- a). Testați dacă R poate ieși din labirint (poate ajunge la margine).
- b). Determinați un drum pentru ieșire (dacă există).

Indicație

Fie T mulțimea pozițiilor în care robotul poate ajunge pornind de la poziția inițială. Notăm cu S mulțimea pozițiilor în care robotul a ajuns până la un moment dat și din care s-ar putea deplasa. Un algoritm pentru determinarea mulțimilor T și S ar putea fi:

```

T ← {poziția inițială}
S ← {poziția inițială}
Cât-timp S ≠ ∅ execută
    Fie p un element din S
    S ← S \ {p}
    Pentru fiecare poziție q alăturată poziției p, q ≠ 'X' și q ∉ T execută
        S ← S ∪ {q}
        T ← T ∪ {q}
    SfPentru
SfCatTimp
    
```

Observații

- Pentru a răspunde la punctul a), algoritmul s-ar putea termina dacă poziția q care satisface condițiile este pe frontiera labirintului

- Structura S poate fi o Stivă sau o Coadă (pentru a răspunde la a) și b)), Coadă (pentru a răspunde la c))
- Mulțimea T poate fi memorată printr-o matrice asociată labirintului (ex: 0 pentru pozițiile neatinse încă, respectiv 1 pentru pozițiile în care robotul a ajuns).
- Pentru a răspunde la punctul b), ne putem gândi la un algoritm care pornind de la o poziție de pe frontieră (în cazul în care răspunsul la punctul a) a fost pozitiv) merge din aproape în aproape pe pozițiile marcate cu 1 (atinse) spre poziția inițială.