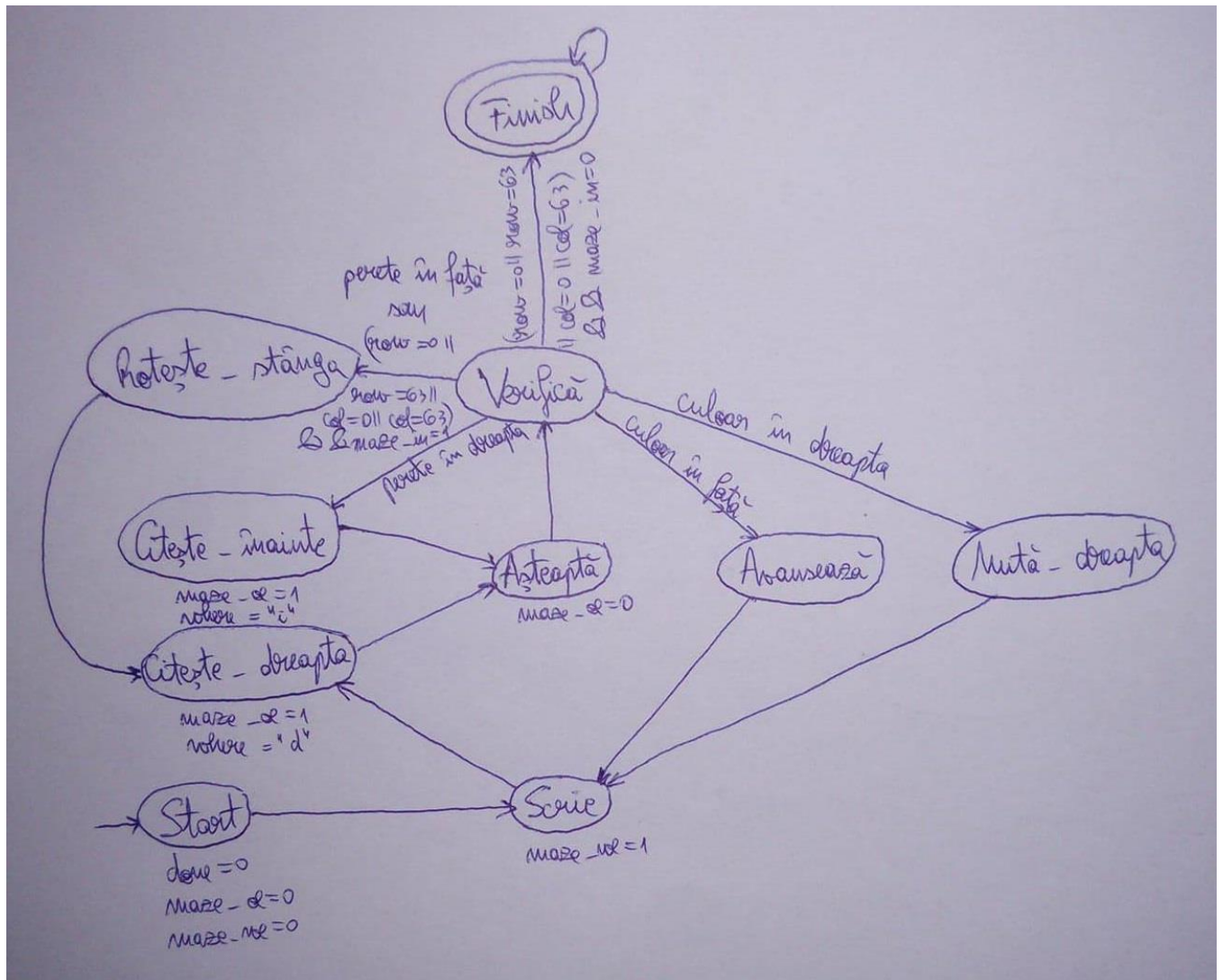


Nițescu Tudor-Alin

331AB

Tema 2 – Maze Solver

Pentru rezolvarea acestei teme, am implementat, așa cum se specifică în cerință, algoritmul Wall Follower cu urmărirea peretelui drept. Pentru acest lucru, am conceput un automat finit cu 10 stări și l-am implementat ulterior. Schema automatului este următoarea:



Logica pe care am urmărit-o în implementarea acestui automat a fost următoarea: atât timp cât cu mâna dreaptă ating peretele din dreapta și am culoar în față, avansează, dacă nu mai ating peretele din dreapta(am culoar în dreapta), mă rotesc spre dreapta 90 de grade și avansează o poziție. Dacă avansarea mea este blocată de perete, mă rotesc spre stânga 90 de grade. Condiția de ieșire din labirint este următoarea: să mă aflu la marginea lui, iar această margine să fie 0(culoar).

Pentru implementarea algoritmică a acestei scheme, în blocul secvențial am făcut trecerea de la o stare la alta, iar în blocul combinațional am implementat fiecare stare a automatului.

Starea inițială a automatului(start) : în această stare se intră doar o singură dată, la începutul algoritmului, și se inițializează poziția curentă cu cea inițială în care ne aflăm în labirint; de asemenea, presupunem că nu ne aflăm la finalul labirintului, adică done=0. După inițializare, intrăm în scriere, pentru a actualiza labirintul.

Starea citeste dreapta: citim o căsuță la dreapta, față de direcția în care ne uităm. Pentru a implementa acest lucru, am utilizat un bloc case, după direcție(sus, jos, stânga, dreapta) pentru cele 4 variante: dacă mă uit în jos, poziția din dreapta este o coloană spre stânga labirintului și rândul curent, dacă mă uit în sus, poziția din dreapta este o coloană spre dreapta labirintului și rândul curent, dacă mă uit spre dreapta, poziția din dreapta este un rand mai jos și coloana curentă, dacă mă uit spre stânga, poziția din dreapta este un rand mai sus și coloana curentă. maze\_oe = 1, pentru a putea citi din labirint. După această citire, reținem că ultima dată am făcut citirea o căsuță la dreapta(where = d) și trecem în starea de așteptare.

Starea citeste inainte: similară stării citeste\_dreapta, doar că vom citi o casuță înainte față de direcția în care ne uităm. Și aici, maze\_oe = 1, pentru a permite citirea din labirint. După această citire, reținem că ultima dată am făcut citirea o căsuță în față(where = i) și trecem în starea de așteptare.

Starea asteapta: stare în care intrăm din stările de citire și în care așteptăm un clock și actualizăm maze\_oe cu 0 pentru a termina citirea propriu-zisă din labirint. Din această stare intrăm în starea de verificare a căsuței citite anterior.

Starea verifica: verificăm întâi dacă am ajuns la finish(dacă ne aflăm în marginea labirintului, iar această margine e culoar). Dacă această margine e culoar, marcăm finalul labirintului prin done=1 și trecem în starea de finish, unde vom rămâne mereu după aceea. Dacă ne aflăm la o poziție de margine, iar marginea este zid, ne rotim spre stânga și continuăm căutarea marginii culoar a labirintului. Dacă ultima căsuță citită a fost în dreapta: dacă această căsuță este culoar, ne rotim spre dreapta și avansăm o poziție. Dacă această căsuță este zid, citim o căsuță în față. Dacă ultima căsuță citită a fost înainte: dacă această căsuță e culoar, avansăm o poziție, altfel ne rotim spre stânga.

Starea avanseaza: în această stare avansăm o poziție înainte față de direcția în care ne uităm(implementare cu un bloc case, după direcție). După ce am avansat, actualizăm labirintul intrând în starea scrie.

Starea muta dreapta: în această stare intrăm în momentul în care nu mai avem perete în dreapta, și trebuie să ne rotim în dreapta și să avansăm o poziție pentru a avea din nou perete în dreapta (implementare cu un bloc case, după direcție). După ce am făcut acest lucru, actualizăm labirintul intrând în starea scrie.

Starea rotește stanga: in această stare intrăm când în față avem zid și trebuie să schimbăm direcția spre care ne uităm, pentru a putea avansa(implementare cu un bloc case, după direcție; \*observație: dacă am dat de o fundătură, ne vom roti spre stânga de 2 ori, pentru a ne întoarce) sau suntem la o poziție de margine, iar marginea respectivă este zid. După ce am schimbat reperul de orientare, citim din nou o căsuță la dreapta față de noul reper(intram in starea citeste dreapta).

Starea scrie: utilizată pentru actualizarea labirintului. Pentru acest lucru, este necesară atribuirea maze\_we=1, pentru a putea scrie în labirint. După actualizare, citim din nou o căsuță la dreapta și reluăm algoritmul.

Starea finish: aici ajungem când suntem în marginea labirintului(done=1), iar această margine e culoar. Din această stare nu ne vom mai muta nicăieri, deoarece am găsit finalul labirintului.