Домашна задача 1
Група 2

1.Проблем со пребарување на простор на состојби

Токсичен Пакман

(а)Минималната репрезентација на состојбата на овој проблем со (k) број пакмани би ја дефинирал со:

- k парови од координати (X , Y) за секој пакмен посебно за да ја чуваме нивната местоположба во матрицата.
- k торки кои во себе ќе имаат од 0 до 3 парови на координати ((x1,y1), (x2,y2), (x3,y3)) кои всушност ги претставуваат координатите на токсичните полиња кои ги оставаат соодветните пакмени позади себе.

Причината поради која не чувам MxN полиња со булеан вредност е поради тоа што треба да се памти кое поле треба да се избрише наредната итерација.

(б) Може да има најмногу $(M*N)^k * (M*N)^{3k} = (M*N)^{4k}$, односно различните позиции на пакмените во матрица/координатен систем од M*N полиња и различните комбинации на секоја од позициите на која може да се наоѓаат нивните токсични полиња.

(в) Максималниот фактор на разгранување, доколку сметаме дека секоја наредна состојба е една од 5 акции кај секој од пакманите („чекај“ , „лево“, „десно“, „горе“, „доле“) ќе биде 5^k доколку секој од нив се движат истовремено.

(г)Доколку го гледаме „лавиринтиот“ како координатен систем,
почетната состојба според мојата дефиниција би била:

(((0,1)(0,4)),((,) (,))),

Каде во првата торка од состојбата, се чуваат двете координати на пакманите , а во втората се чуваат координатите на нивните токсични полиња соодветно.

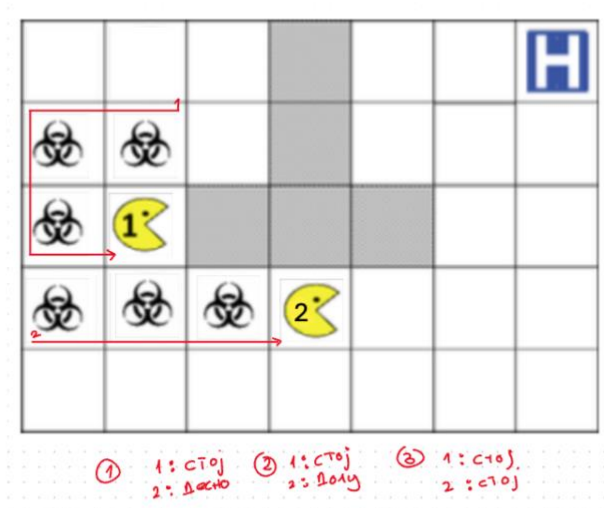
Целната состојба на проблемот би била:

$$(((6,4),(6,4)),(((X_1,Y_1), (X_2,Y_2), (X_3,Y_3)), ((X_1,Y_1), (X_2,Y_2), (X_3,Y_3)))),$$

Каде во првата торка во состојбата се чуваат позициите на пакманите кои влегле во болницата а во втората се чуваат нивните координати на токсични полиња соодветно, меѓутоа бројката на токсичните полиња и нивниите координати не се важни за состојбата доколку таа е целна, и тие не ни значат.

(д)Доколку ја имаме следната произволна состојба, постојат 3 дозволени акции, поточно:
- 1: чекај 2:десно
- 1: чекај 2:долу
- 1: чекај 2: чекај

Пакманот со број 1 , во било која акција од моменталната состојба не би можел да се придвижи никаде поради тоа што има забранети полиња околу себе и ќе мора да чека токсично поле да се ослободи, додека пакманот со број 2 има три опции , чекај, десно и доле и затоа од оваа произволна состојба имаме три акции односно се разгранува на 3 нови состојби во графот на состојби.



(ѓ)Една не-тривијална и допустлива хевристика за сценарио со еден пакман би била неговото „менхетен“ растојание до болницата (целта), односно $abs(x_p - x_h) + abs(y_p - y_h)$, каде p е соодветната координата на пакменот , а h на куќичката. Оваа хевристика е допустлива поради тоа што на која било позиција да се наоѓа пакменот, најкраткото растојание до целта би му било менхетен растојанието. Никогаш не би можела оваа хевристика да врати вредност поголема од вистинското растојание од пакменот до куќата, поради тоа што таа го пресметува најкраткиот пат во оптимистичен случај, без препреки по патот (зид или токсично поле), додека во реалноста многу веројатно е дека ќе ги има по патот, но и да ги нема сепак не може да стигне пакманот со цена помала од таа што ќе ја врати оваа хевристика.

(е)

1) $h_a: \frac{\sum_{i=1}^{k} h_i}{k}$ 2) $h_b: \sum_{i=1}^{k} h_i$ 3) $h_c: \max_{1 \le i \le k} h_i$ 4) $h_d: k * \max_{1 \le i \le k} h_i$ 5) $h_e: \min_{1 \le i \le k} h_i$ 6) $h_f: k * \min_{1 \le i \le k} h_i$

1) Оваа хевристика ни дава просек од менхетен растојанијата на сите пакмани до болницата и би била допустлива во секој случај поради тоа што има пакмани

што растојанието им е над-просекот и реалното решение секогаш би било поголемо од хевристиката.

2) Оваа хевристика враќа збир од сите менхетен растојанија на пакманите , ова не е допуслива хевристика поради тоа што сите пакмани се движат истовремено, да беше случај да се движеа едно по едно би била валидна хевристика, но во овој случај пакманите ќе стигнат сигурно со цена помала од оваа хевристика бидејќи оваа хевристика како еден од операндите (најголемиот менхетен од пакманите) може да ја содржи и реалната цена што значи дека $h_b > h^*_b$.
3) Оваа хевристика е допустлива поради тоа што реалната цена во оптимистичен случај, никој пакман да нема препреки по пат, ќе биде еднаква на цената на најодалечениот пакман, што всушност е излезот на оваа хевристика. Тоа значи дека $h_c = h^*_c$ што е во ред поради тоа што бараме хевристика која е помала или еднаква од реалната цена.
4) Оваа хевристика, доколку го имаме во предвид излезот од претходната (3) , сигурни сме дека во сите случаи враќа цена поголема(или еднаква ако е 0) од реалната цена што значи дека не е допустлива.
5) Оваа хевристика го дава минималното растојание од сите пакмени до болницата , што значи дека $h_e \leq h_c => h_e \leq h^*$ односно оваа хевристика сигурно ќе биде помала или еднаква од реалната цена
6) Оваа хевристика не е допустлива поради тоа што ќе постои случај кога k помножено со минималното растојание ќе биде поголемо од реалната цена. Пример за тоа е кога сите растојанија на пакмените се исти и минимумот на растојанијата е всушност исто и со максимумот и тогаш го имаме истиот случај како во 4) , знаеме со сигурност дека излезот ќе биде поголем од реалната цена.

(ж) DFS алгоритамот нема да работи со хевристики и во просечен случај не би бил соодветен бидејќи не е оптимален,а исто така не користи хевристика со што преработуваме во грешна насока голем дел од случаите. Тој не би вратил најбрзо решение , но би вратил првото што ќе го најде.
BFS и UCS во овој случај ќе се однесуваат исто поради тоа што цената на патот на секој од пакмените во овој пример е иста , така што тие би биле оптимални и би го вратиле најбрзото решение, но сепак ќе се обработуваат и голем број непотребни јазли што е резултат на не-користењето на хевристика.
A* алгоритамот би дало најоптимално и најбрзо решение поради тоа што го насочува BFS(или UCS) кон вистинското решение, но за тоа е потребна допустлива, конзистентна и не-тривијална хевристика која ќе му помогне побргу да стине до целта.

2. Проблем кој задоволува услови

Какурасу

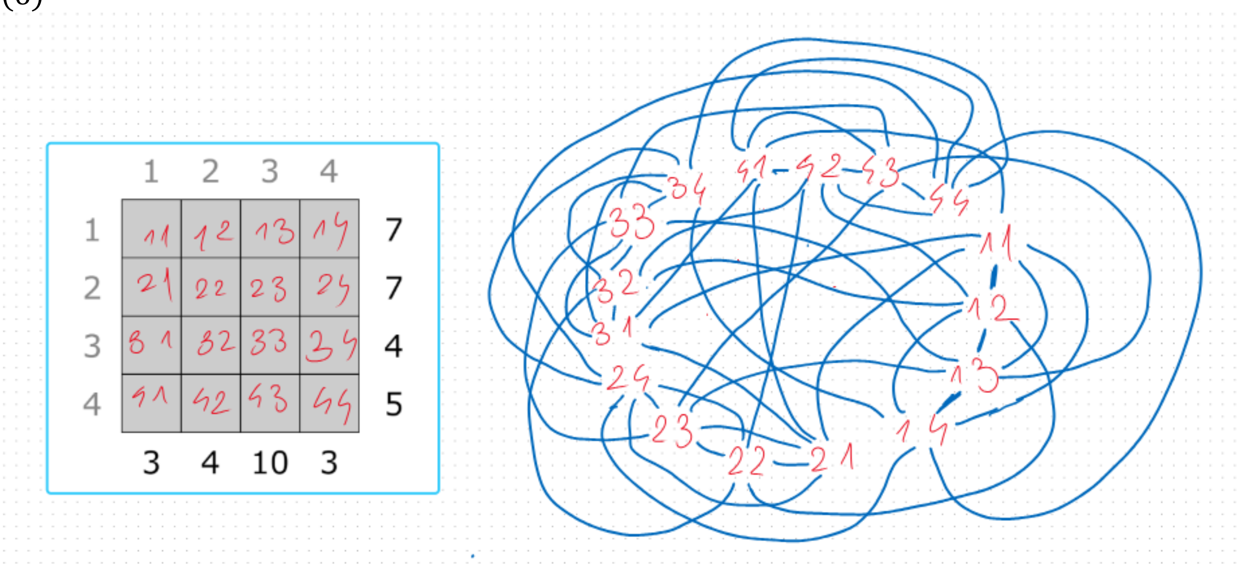
(а) За овој проблем на исполнување услови би ги дефинирале :

-Променливите:
Листа од полиња на дел од матрица која започнува со индексите (1,1) наместо (0,0).
Таа ќе изгледа вака:
(1, 1), (2, 1), (3, 1), (4, 1), (1, 2), (2, 2), (3, 2), (4, 2), (1, 3), (2, 3), (3, 3), (4, 3), (1, 4), (2, 4), (3, 4), (4, 4)
-Домените кои ќе ги доделуваме на овие променливи:

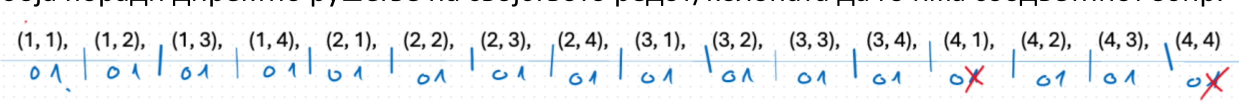
    0 – необоено поле
    1 – обоено поле во сино

-Ограничивања:
За секоја координата во една колона, домените на истата да бидат доделени така што збирот на y координатите на доделените полиња да биде еднаков на сумата за таа колона.
Исто и за редиците,
За секоја координата во една редица, домените на истата да бидат доделени така што збирот на x координатите на доделените полиња да биде еднаков на сумата за таа редица.

(б)



(в) Во овој проблем можеме, но не мораме да поставиме унарни услови за променливите кои имаат вредност поголема од целниот збир во нивната колона/редица да не смеат да бидат обоени со сина боја. Доколку го имлементираме тоа, ќе ги отстраниме вредностите за сина боја од домените на 41 и 44 поради тоа што нема да смее во нив да се чува сина боја поради директно рушење на својството редот/колоната да го има соодветниот збир.

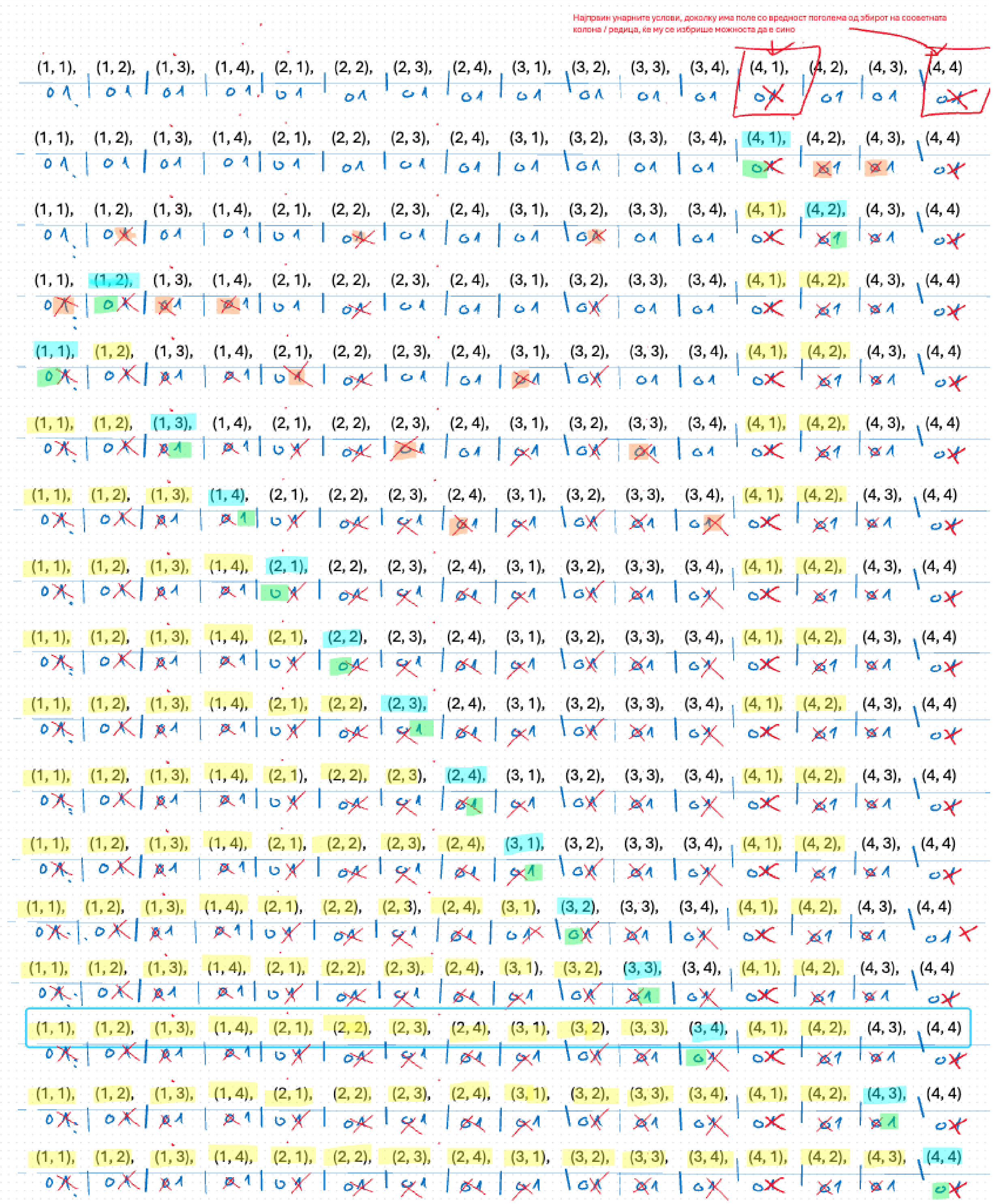


(г)

Во следната слика правам проверка на конзистентноста на сите ребра, каде што за секоја променлива што е кај опашот на реброто и не е конзистентна ја имам заокружено и ѝ имам одземено една вредност од доменот која не е конзистентна во однос на променливата кај главата на реброто. Ова значи дека го отстранувам доменот на една променлива од опашот во ситуацијата: доколку се одбере тој домен , нема да може да се одбере домен кај променливата кај главата на реброто за да се исполни условот за збир на обоените координати во соодветната колона/редица.

(д) Во следната слика најдов едно решение на проблемот користејќи ги MRV и LCV хевристиките и алгоритамот forward checking.
Одбирањето на променлива според MRV е претставено со светло сина боја, одбирањето на домен од променлива според LCV е претставено со зелена боја, а бришењето на невалидните домени според алгоритамот forward checking е претставено со портокалова боја.
Со жолта боја се претставени веќе поминатите променливи на кои им е доделен домен.
Бирањето на променлива и домен според хевристиките влијае врз други променливи односно нивните домени и за тоа се грижи алгоритамот за враќање наназад со проверка нанапред. На пример вториот ред од сликата односно по првото

бирање на променлива и домен ((4,1) со домен 0) , може да се забележи дека бирањето на оваа променлива директно влијае врз променливите (4,2) и (4,3) поради тоа што доколку (4,1) не е обоено, во таа редица постои само една можна комбинација на обоени координати односно парот (4,2) и (4,3) и затоа нивните домени за да бидат необоени се отстранети.



Изработено од:
Горазд Филиповски,
223070