

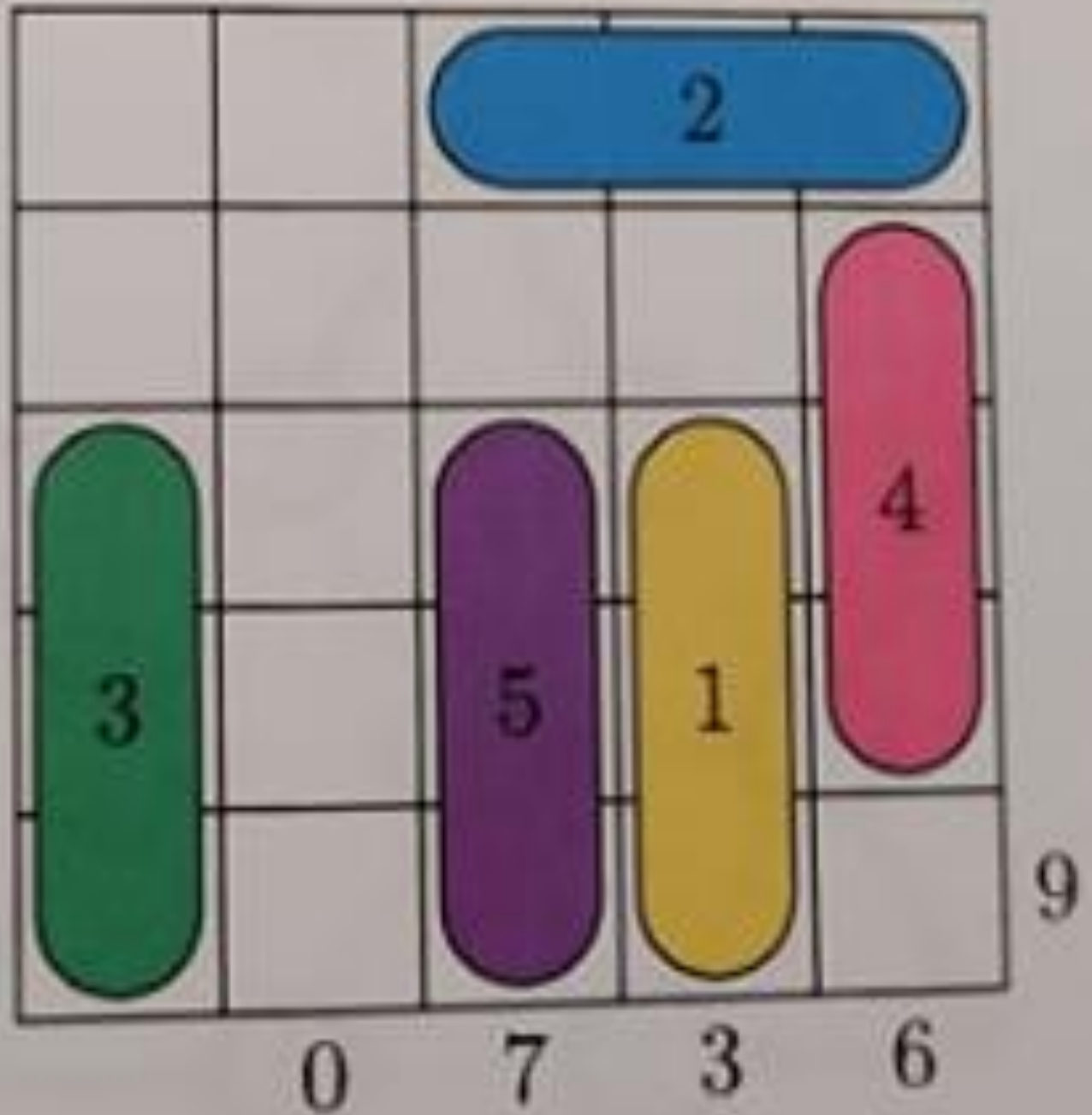
Mesterséges intelligencia alkalmazások

Csatahajók

Harsányi Tibor

Feladat

- Adott öt darab, 1-től 5-ig számozott 3x1-es méretű torpedóromboló, amelyet elrejtettünk egy 5x5-ös négyzetrácson. A négyzetrács oldalain, a sorok és oszlopok mellett található számok jelzik az adott sorban vagy oszlopban lévő hajók sorszámainak összegét.
- A feladat: elhelyezni a torpedórombolókat



Állapottér

Egy állapot az alábbi módon nézhet ki:

$a_{1,1}$	$a_{1,2}$	$a_{1,3}$	$a_{1,4}$	$a_{1,5}$	s_1
$a_{2,1}$	$a_{2,2}$	$a_{2,3}$	$a_{2,4}$	$a_{2,5}$	s_2
$a_{3,1}$	$a_{3,2}$	$a_{3,3}$	$a_{3,4}$	$a_{3,5}$	s_3
$a_{4,1}$	$a_{4,2}$	$a_{4,3}$	$a_{4,4}$	$a_{4,5}$	s_4
$a_{5,1}$	$a_{5,2}$	$a_{5,3}$	$a_{5,4}$	$a_{5,5}$	s_5
o_1	o_2	o_3	o_4	o_5	

- $a_{i,j}$ ($i, j \in \{1, 2, 3, 4, 5\}$) jelöli, a pozíciókban lévő torpedóhajók értékét
 $a_{i,j} \in \{0, 1, 2, 3, 4, 5\}$
- o_i ($i \in \{1, 2, 3, 4, 5\}$) jelöli, hogy az adott oszlopban még mennyi értékű torpedóhajó tehető le
 $o_i \in \{\text{null}, 0, 7, 3, 6\}$
- s_i ($i \in \{1, 2, 3, 4, 5\}$) jelöli, hogy az adott sorban még mennyi értékű torpedóhajó tehető le
 $s_i \in \{\text{null}, 9\}$
- A -1 érték azt jelzi, hogy az adott sorban vagy oszlopban nincs kikötés a torpedóhajó értékére vonatkozóan
- $\text{lerakhato} = \{1, 2, 3, 4, 5\}$ a még lerakható hajók halmaza

Jelölje „a” $a(z)$

$$\left(\begin{pmatrix} a_{11} & \cdots & a_{15} \\ \vdots & \ddots & \vdots \\ a_{51} & \cdots & a_{55} \end{pmatrix}, (s_1, s_2, s_3, s_4, s_5), (o_1, o_2, o_3, o_4, o_5), \text{lerakhato} \right)$$

állapotot

Ekkor a probléma állapottere:

$$A = \{ a \mid a_{ij} \in \{0,1,2,3,4,5\} \wedge s_k \in \{\text{null}, 1,2,3,4,5,6,7,8,9\} \wedge o_z \in \{\text{null}, 0,1,2,3,4,5,6,7\} \wedge (i,j,k,z) \in \{1,2,3,4,5\} \wedge \text{lerakhato} = \{1,2,3,4,5\} \}$$

Kezdőállapot

$$\text{Kezdő} = \left(\begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}, (-1, -1, -1, -1, 9), (-1, 0, 7, 3, 6), \{1, 4, 5\} \right)$$

Célállapotok halmaza:

Egy állapot akkor lesz célállapot ha az összes hajó le van helyezve. Ez akkor valósul meg ha az a_{ij} ($i, j \in \{1, 2, 3, 4, 5\}$) pozíciókban lévő számok összege 45 mivel minden számból 3db lesz elhelyezve. Ezen felül a sorokban és oszlopokban lehető értékek null vagy 0 azaz megvan minden sorban és oszlopban a megfelelő értékű torpedóhajó.

$$\text{Cél} = \{c \mid c \in A \wedge \text{lerakhato} = \{\} \wedge \forall k (s_k = -1 \vee s_k = 0) \wedge \forall z (o_z = -1 \vee o_z = 0)\}$$

Operátorok halmaza:

$$O = \{v_lerak_{i,j,T}(a), f_lerak_{i,j,T}(a)\}$$

ahol a v_lerak a vízszintesen való elhelyezés, az f_lerak pedig a függőlegesen való elhelyezés.

Operátor alkalmazási előfeltételek:

$$\text{előfeltétel_v_lerak}_{i,j,T}(a) = (T \in \text{lerakhato} \wedge (1 \leq i \leq 5) \wedge (1 \leq j \leq 3) \wedge (s_i = -1 \vee s_i - T \geq 0) \wedge (o_i = -1 \vee o_i - T \geq 0) \wedge (o_{j+1} = -1 \vee o_{j+1} - T \geq 0) \wedge (o_{j+2} = -1 \vee o_{j+2} - T \geq 0) \wedge (a_{i,j} = 0) \wedge (a_{i,j+1} = 0) \wedge (a_{i,j+2} = 0))$$

$$\text{előfeltétel_f_lerak}_{i,j,T}(a) = (T \in \text{lerakhato} \wedge (1 \leq i \leq 3) \wedge (1 \leq j \leq 5) \wedge (o_i = -1 \vee o_i - T \geq 0) \wedge (s_i = -1 \vee s_i - T \geq 0) \wedge (s_{i+1} = -1 \vee s_{i+1} - T \geq 0) \wedge (s_{i+2} = -1 \vee s_{i+2} - T \geq 0) \wedge (a_{i,j} = 0) \wedge (a_{i+1,j} = 0) \wedge (a_{i+2,j} = 0))$$

i,j jelöli a pozíciót ahova leszeretnénk tenni a hajót T pedig az adott értékű hajót jelöli $T \in \text{lerakhato}$

Operátor hatásdefiníció:

$$v_lerak_{i,j,T}(a) = a'$$

$$i,j,x,y \in \{1,2,3,4,5\}$$

$$f_lerak_{i,j,T}(a) = a'$$

$$i,j,x,y \in \{1,2,3,4,5\}$$

$$v_{\text{lerak}_{i,j,T}}(a) = a'$$

$$i,j,x,y \in \{1,2,3,4,5\}$$

$$s'_x = \begin{cases} s_i - T, & \text{ha } x = i \wedge s'_x \neq -1 \\ s'_x & \text{egyébként} \end{cases}$$

$$a'_{xy} = \begin{cases} T, & \text{ha } x = i \wedge y = j \\ T, & \text{ha } x = i \wedge y = j + 1 \\ T, & \text{ha } x = i \wedge y = j + 2 \\ a'_{xy} & \text{egyébként} \end{cases}$$

$$o'_y = \begin{cases} o_j - T, & \text{ha } y = j \\ o_{j+1} - T, & \text{ha } y = j + 1 \\ o_{j+2} - T, & \text{ha } y = j + 2 \\ o'_y & \text{egyébként} \end{cases}$$

$$\text{lerakhato} = \text{lerakhato} \setminus \{T\}$$

$$f_lerak_{i,j,T}(a) = a'$$

$$i,j,x,y \in \{1,2,3,4,5\}$$

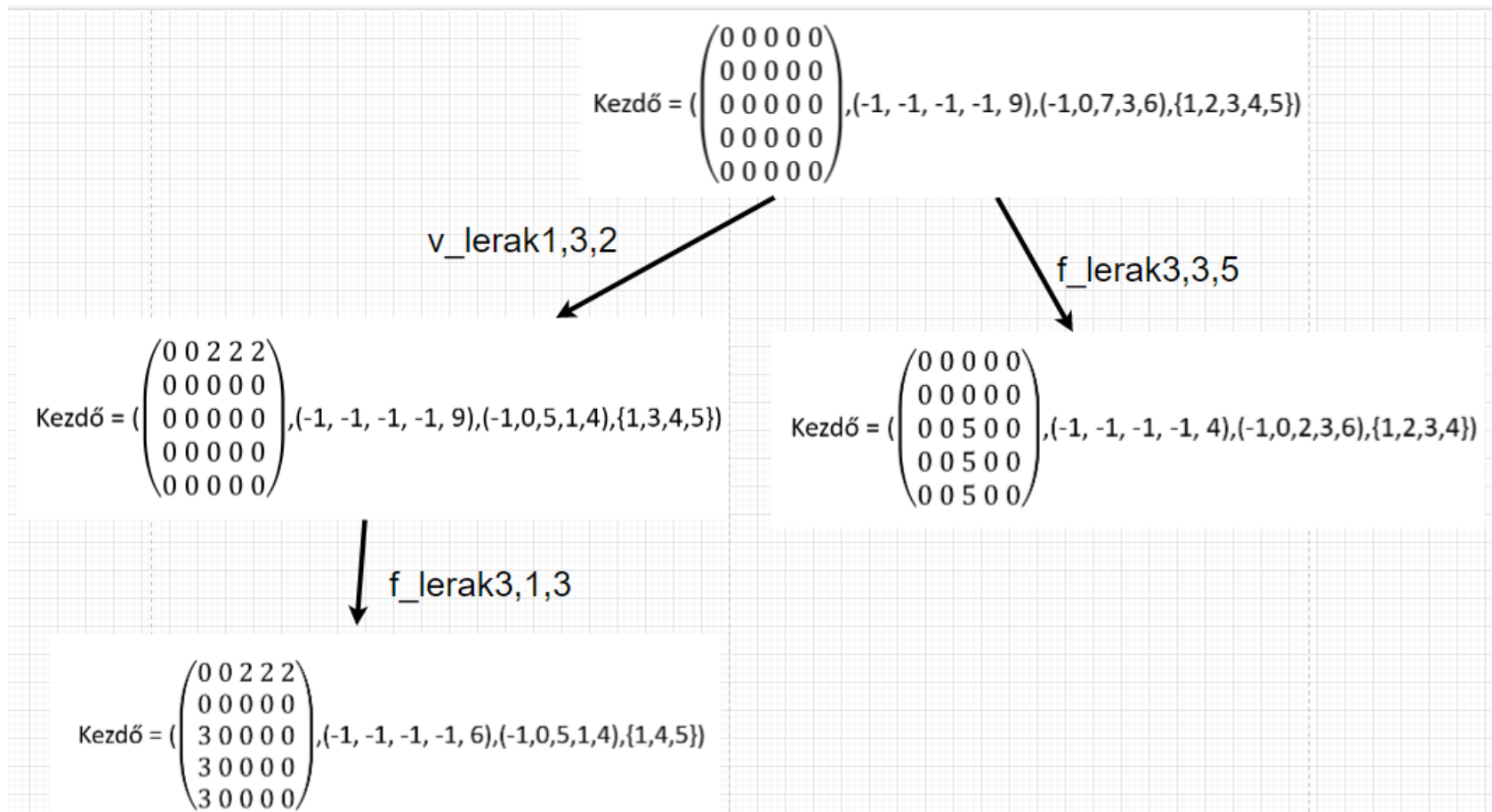
$$o'_y = \begin{cases} o_j - T, ha\ y = j \wedge o'_y \neq -1 \\ o'_y\ egyébként \end{cases}$$

$$a'_{xy} = \begin{cases} T, ha\ x = i \wedge y = j \\ T, ha\ x = i + 1 \wedge y = j \\ T, ha\ x = i + 2 \wedge y = j \\ a'_{xy}\ egyébként \end{cases}$$

$$s'_x = \begin{cases} s_i - T, ha\ x = i \\ s_{i+1} - T, ha\ x = i + 1 \\ s_{i+2} - T, ha\ x = i + 2 \\ s'_x\ egyébként \end{cases}$$

$$lerakhato = lerakhato \setminus \{T\}$$

Gráf részlet:



Kereső algoritmus

- A kereséshez mélységi keresőt választottam mivel megoldás mindig a lehető legmélyebben található.
- Ez abból következik, hogy 5 lerakható hajó van és 5 operátor alkalmazásával el is fogynak az alkalmazható operátorok.
- És tudjuk hogy mind az 5 hajónak le kell téve lenniük, hogy a célfeltétel egy része teljesülhessen.

```

for (int i = 0; i < 5; i++) {
    for (int j = 0; j < 5; j++) {
        for (int t = 1; t <= 5; t++) {
            operators.add(new Operator(i, j, t, "v"));

            operators.add(new Operator(i, j, t, "f"));
        }
    }
}

```

- Az alábbi két for ciklussal létrehozom az összes operátort
- Olyan operátorok is létre fognak jönni amelyek nem alkalmazhatóak mivel ha az $i = 5$ és $j = 5$ akkor az operátor az 5,6,7 pozícióra szeretne elhelyezni egy hajót amely nem lehetséges de ezt majd az alkalmazási előfeltétel kiküszöböli

```

public static void melysegi(List<Operator> operators, State kezdo){

    Csomopont newCsomopont = new Csomopont(kezdo,null,null,0);
    nyiltak.add(newCsomopont);
    zartak = new HashSet<>();
    Csomopont csomopont = new Csomopont();

    while(true){

        if(nyiltak.isEmpty()){
            break;
        }

        csomopont = nyiltak.stream().max(Comparator.comparing(Csomopont::getMelyseg)).orElseThrow();

        if(csomopont.getState().cel()){
            break;
        }
        kiterjeszt(csomopont,operators);
    }
    if(!nyiltak.isEmpty()){
        megoldasKilir(csomopont);
    }
    else{
        System.out.println("Nincs megoldas");
    }
}

```

- Ezen a képen a mélységi kereső látható
- A kijelölt rész fogja kiválasztani a legnagyobb mélységű nyílt csomópontot

Köszönöm a figyelmet !