

Dvojrozmerné pole

Úloha

Dané je dvojrozmerné pole a , ktorého prvky sú $a = \begin{pmatrix} -2 & 3 & 5 \\ -1 & -3 & 8 \\ 1 & -5 & -9 \\ -2 & 2 & 4 \end{pmatrix}$.

Nájdite (ak sa dá aspoň tromi spôsobmi)

- počet prvkov poľa a , počet riadkov v poli, jeho dimenziu
- vypíšte prvky ležiace v treťom riadku,
- vypíšte prvky ležiace v druhom stĺpci
- vypíšte (aj v maticovom tvare) podpole, ktoré je tvorené prvkami z prvých troch riadkov a zároveň z prvých dvoch stĺpcov
- vypíšte (aj v maticovom tvare) podpole, ktoré je tvorené prvkami z posledných dvoch riadkov a zároveň z prvých dvoch stĺpcov
- vypíšte prvky $a_{1,3}$, $a_{2,1}$
- vypíšte prvok, ktorý leží v štvrtom riadku a druhom stĺpci a prvok, ktorý leží v treťom stĺpci a druhom riadku
- do jednorozmerného poľa b vyberte a zapíšte druhý a tretí riadok poľa a
- miesto (pozíciu), kde sa v poli nachádza prvok -5
- maximálny a minimálny prvok
- súčet všetkých prvkov a aritmetický priemer všetkých prvkov
- súčet prvkov v treťom riadku
- súčet prvkov v druhom stĺpci a ich priemer
- počet prvkov z poľa a , ktorých hodnota je menšia ako $+4$
- súčet prvkov z poľa a , ktorých hodnota je menšia ako $+4$
- súčet prvkov z poľa a , ktorých hodnota je menšia ako $+4$ a ležia v druhom stĺpci
- počet prvkov z poľa a , ktorých hodnota je menšia ako $+4$ a ležia v treťom alebo vo štvrtom riadku
- počet prvkov z poľa a , ktorých hodnota je záporná a ležia v druhom alebo v treťom stĺpci
- vypíšte všetky párne prvky v poli
- vypíšte všetky záporné nepárne prvky
- nahraďte tretí riadok samými dvojkami
- nahraďte prvky poľa, ktoré ležia v treťom a štvrtom riadku a v druhom a treťom stĺpci nulami
- odstráňte štvrtý riadok matice
- pridajte na začiatok riadok, ktorý je tvorený prvkami $\{5, 7, 9\}$

Riešenie

```
In[470]:= Clear[A]
A = {{-2, 3, 5}, {-1, -3, 8}, {1, -5, -9}, {-2, 2, 4}}
Out[471]= {{-2, 3, 5}, {-1, -3, 8}, {1, -5, -9}, {-2, 2, 4}}
```

Počet prvkov pol'a A , počet riadkov v poli, jeho dimenziu

```
In[472]:= Flatten[A] // Length
Out[472]= 12

In[473]:= Length[A]
Out[473]= 4

In[474]:= Dimensions[A]
Out[474]= {4, 3}
```

Vypíšte prvky ležiace v treťom riadku

```
In[475]:= A[[3]]
Out[475]= {1, -5, -9}

In[476]:= A[[-2]]
Out[476]= {1, -5, -9}

In[477]:= Part[A, 3]
Out[477]= {1, -5, -9}
```

Vypíšte prvky ležiace v druhom stĺpci

```
In[478]:= Transpose[A][[2]]
Out[478]= {3, -3, -5, 2}

In[479]:= A[[1 ;; 4, 2]]
Out[479]= {3, -3, -5, 2}

In[480]:= A[[All, 2]]
Out[480]= {3, -3, -5, 2}
```

Vypíšte (aj v maticovom tvare) podpole, ktoré je tvorené prvkami z prvých troch riadkov a zároveň z prvých dvoch stĺpcov

```
In[481]:= A[[1 ;; 3, 1 ;; 2]]
Out[481]= {{-2, 3}, {-1, -3}, {1, -5}}
```

```
In[482]:= % // MatrixForm
Out[482]//MatrixForm=
```

$$\begin{pmatrix} -2 & 3 \\ -1 & -3 \\ 1 & -5 \end{pmatrix}$$

Vypíšte (aj v maticovom tvare) podpole, ktoré je tvorené prvkami z posledných dvoch riadkov a zároveň z prvých dvoch stĺpcov

```
In[483]:= A[[-2 ;; -1, 1 ;; 2]]
Out[483]= {{1, -5}, {-2, 2}}
```

Poznámka : keby sme chceli aj otočiť poradie riadkov

```
In[484]:= A[[-1 ;; -2 ;; -1, 1 ;; 2]]
Out[484]= {{-2, 2}, {1, -5}}
```

Vypíšte prvky $a_{1,3}$, $a_{2,1}$

```
In[485]:= A[[1, 3]]
Out[485]= 5
```

```
In[486]:= A[[2, 1]]
Out[486]= -1
```

Vypíšte prvok, ktorý leží v štvrtom riadku a druhom stĺpci a prvok, ktorý leží v treťom stĺpci a druhom riadku

```
In[487]:= A[[4, 2]]
Out[487]= 2
```

```
In[488]:= A[[2, 3]]
Out[488]= 8
```

Do jednorozmerného poľa b vyberte a zapíšte druhý a tretí riadok poľa A

```
In[489]:= A[[2]]
Out[489]= {-1, -3, 8}
```

```

In[490]:= A[[3]]
Out[490]= {1, -5, -9}

In[491]:= b = {A[[2]], A[[3]]} // Flatten
Out[491]= {-1, -3, 8, 1, -5, -9}

In[492]:= b = Append[A[[2]], A[[3]]] // Flatten
Out[492]= {-1, -3, 8, 1, -5, -9}

```

Určite miesto (pozíciu), kde sa v poli nachádza prvok -5

```

In[493]:= Position[A, -5]
Out[493]= {{3, 2}}

```

Určite miesto (pozíciu), kde sa v poli nachádza maximálny a minimálny prvok

```

In[494]:= Max[A]
Out[494]= 8

In[495]:= Position[A, Max[A]]
Out[495]= {{2, 3}}

In[496]:= Min[A]
Out[496]= -9

In[497]:= Position[A, Min[A]]
Out[497]= {{3, 3}}

```

Súčet všetkých prvkov a aritmetický priemer všetkých prvkov

1. spôsob - súčet

```

In[498]:= Sum[A[[i, j]], {i, 1, 4}, {j, 1, 3}]
Out[498]= 1

```

2. spôsob - súčet

```

In[499]:= Sum[Sum[A[[i, j]], {j, 1, 3}], {i, 1, 4}]
Out[499]= 1

```

3. spôsob - súčet

```

In[500]:= Apply[Plus, A // Flatten]
Out[500]= 1

```

4. spôsob - súčet

```

In[501]:= sucet = 0;
Do[
  Do[
    sucet = sucet + A[[i, j]], {i, 1, 4}
  ],
  {j, 1, 3}]
sucet

Out[503]= 1

```

5. spôsob - súčet

```

In[504]:= Total[A // Flatten]

Out[504]= 1

```

Aritmetický priemer všetkých prvkov - ukážeme si niekoľko spôsobov ako danú úlohu naprogramovať

1. spôsob - aritmetický priemer

```

In[505]:= 1 / Length[A // Flatten] * Sum[A[[i, j]], {i, 1, 4}, {j, 1, 3}]

Out[505]=  $\frac{1}{12}$ 

```

2. spôsob - aritmetický priemer

```

In[506]:= 1 / 12  $\sum_{i=1}^4 \sum_{j=1}^3 A[[i, j]]$ 

Out[506]=  $\frac{1}{12}$ 

```

3. spôsob - aritmetický priemer

```

In[507]:= Mean[A // Flatten]

Out[507]=  $\frac{1}{12}$ 

```

4. spôsob - aritmetický priemer

```

In[508]:= 1 / Length[A // Flatten] * Apply[Plus, A // Flatten]

Out[508]=  $\frac{1}{12}$ 

```

Súčet prvkov v treťom riadku

```
In[509]:= Total[A[[3]]]
```

```
Out[509]= -13
```

Súčet prvkov v druhom stĺpci a ich priemer

```
In[510]:= Total[A[[1 ;; 4, 2]]]
```

```
Out[510]= -3
```

```
In[511]:= Mean[A[[1 ;; 4, 2]]]
```

```
Out[511]=  $-\frac{3}{4}$ 
```

Počet prvkov z poľa A, ktorých hodnota je menšia ako + 4

1. spôsob - procedurálne programovanie

```
In[512]:= pocet = 0;
```

```
Do[
```

```
    If[A[[i, j]] < 4, pocet ++ ],
```

```
    {i, 1, 4}, {j, 1, 3}];
```

```
pocet
```

```
Out[514]= 9
```

2. spôsob - pomocou pure function

```
In[515]:= b = Select[A // Flatten, # < 4 &]
```

```
Out[515]= {-2, 3, -1, -3, 1, -5, -9, -2, 2}
```

```
Length[b]
```

```
9
```

3. spôsob - pomocou predikátových funkcií

```
In[516]:= testQ[x_] = If[x < 4, True, False];
```

```
Select[A // Flatten, testQ] // Length
```

```
Out[517]= 9
```

4. spôsob - pomocou predikátových funkcií a funkcionálneho programovania

```
In[518]:= testQ[x_] = If[x < 4, True, False];
b = Select[A // Flatten, testQ]
Length[b]
```

```
Out[519]= {-2, 3, -1, -3, 1, -5, -9, -2, 2}
```

```
Out[520]= 9
```

5. spôsob - pomocou zabudovanej funkcie

```
In[521]:= b = Cases[A // Flatten, x_ /; x < 4]
```

```
Out[521]= {-2, 3, -1, -3, 1, -5, -9, -2, 2}
```

```
In[522]:= b // Length
```

```
Out[522]= 9
```

súčet prvkov z poľa *a*, ktorých hodnota je menšia ako + 4

1. spôsob - procedurálne programovanie

```
In[523]:= sucet = 0;
```

```
Do[
```

```
    If[A[[i, j]] < 4, sucet = sucet + A[[i, j]],
```

```
    {i, 1, 4}, {j, 1, 3}];
```

```
sucet
```

```
Out[525]= -16
```

2. spôsob - pomocou pure function

```
In[526]:= b = Select[A // Flatten, # < 4 &]
```

```
Out[526]= {-2, 3, -1, -3, 1, -5, -9, -2, 2}
```

```
In[527]:= Total[b]
```

```
Out[527]= -16
```

3. spôsob - pomocou predikátových funkcií

```
In[528]:= testQ[x_] = If[x < 4, True, False];
```

```
Select[A // Flatten, testQ] // Total
```

```
Out[529]= -16
```

4. spôsob - pomocou predikátových funkcií a funkcionálneho programovania

```
In[530]:= testQ[x_] = If[x < 4, True, False];
```

```
b = Select[A // Flatten, testQ]
```

```
Total[b]
```

```
Out[531]= {-2, 3, -1, -3, 1, -5, -9, -2, 2}
```

```
Out[532]= -16
```

5. spôsob - pomocou zabudovanej funkcie

```
In[533]:= b = Cases[A // Flatten , x_ /; x < 4]
```

```
Out[533]= {-2, 3, -1, -3, 1, -5, -9, -2, 2}
```

```
In[534]:= b // Total
```

```
Out[534]= -16
```

súčet prvkov z poľa a , ktorých hodnota je menšia ako + 4 a ležia v druhom stĺpci

ak máme pracovať s nejakou podmnožinou poľa - NAJSKÔR si ju vytvoríme a potom pracujeme ako obvykle

```
In[535]:= pomocne = A[[All, 2]]
```

```
Out[535]= {3, -3, -5, 2}
```

1. spôsob - procedurálne programovanie

```
In[536]:= sucet = 0;
```

```
Do[
```

```
    If[pomocne[[i]] < 4, sucet = sucet + pomocne[[i]],
```

```
    {i, 1, 4}];
```

```
sucet
```

```
Out[538]= -3
```

2. spôsob - pomocou pure function

```
In[539]:= b = Select[pomocne // Flatten , # < 4 &]
```

```
Out[539]= {3, -3, -5, 2}
```

```
In[540]:= Total[b]
```

```
Out[540]= -3
```

3. spôsob - pomocou predikátových funkcií

```
In[541]:= testQ[x_] = If[x < 4, True, False];
```

```
Select[pomocne // Flatten , testQ] // Total
```

```
Out[542]= -3
```

4. spôsob - pomocou predikátových funkcií a funkcionálneho programovania

```
In[543]:= testQ[x_] = If[x < 4, True, False];
```

```
b = Select[pomocne // Flatten , testQ]
```

```
Total[b]
```

```
Out[544]= {3, -3, -5, 2}
```

```
Out[545]= -3
```

5. spôsob - pomocou zabudovanej funkcie


```
In[546]:= b = Cases[pomocne // Flatten, x_ /; x < 4]
Out[546]= {3, -3, -5, 2}

In[547]:= b // Total
Out[547]= -3
```

- počet prvkov z poľa *a*, ktorých hodnota je menšia ako + 4 a ležia v treťom alebo vo štvrtom riadku

vyberiemie podpole - v prvom kroku bude dvojrozmerné - ale lepšie sa nám ďalej v úlohe bude manipulovať s jednorozmerným poľom a preto ho upravíme.

```
In[548]:= pom1 = A[[3 ;; 4, All]]
Out[548]= {{1, -5, -9}, {-2, 2, 4}}
```

```
In[549]:= pom = pom1 // Flatten
Out[549]= {1, -5, -9, -2, 2, 4}
```

a napríklad jeden zo spôsobov

```
In[550]:= testQ[x_] = If[x < 4, True, False];
Select[pom // Flatten, testQ] // Length
Out[551]= 5
```

- počet prvkov z poľa *a*, ktorých hodnota je záporná a ležia v druhom alebo v treťom stĺpci

```
In[552]:= pom1 = A[[All, 2 ;; 3]]
Out[552]= {{3, 5}, {-3, 8}, {-5, -9}, {2, 4}}

In[553]:= pom = pom1 // Flatten
Out[553]= {3, 5, -3, 8, -5, -9, 2, 4}
```

a napríklad jeden zo spôsobov

```
In[554]:= testQ[x_] = If[x < 0, True, False];
Select[pom // Flatten, testQ] // Length
Out[555]= 3
```

- vypíšte všetky párne prvky v poli

Najjednoduchší spôsob je pomocou príkazu Select, lebo máme predikátorovú testovaciu funkciu EvenQ

```
In[556]:= Select[A // Flatten, EvenQ]
Out[556]= {-2, 8, -2, 2, 4}
```

Ale kto sa nudí, môže to urobiť aj pomocou procedurálneho prístupu

```
In[557]:= parneprvky = {};
pom = A // Flatten ;
Do[If[Mod[pom[[i]], 2] == 0, AppendTo[parneprvky , pom[[i]]],
  {i, 1, Length[pom]}]
parneprvky
Out[560]= {-2, 8, -2, 2, 4}
```

- vypíšte všetky záporné nepárne prvky

Najjednoduchší spôsob je pomocou príkazu Select, lebo máme predikátorovú testovaciu funkciu OddQ a použijeme novú myšlienku z množinovej matematiky

```
In[561]:= pom1 = Select[A // Flatten , OddQ]
Out[561]= {3, 5, -1, -3, 1, -5, -9}

In[562]:= pom2 = Select[A // Flatten , # < 0 &]
Out[562]= {-2, -1, -3, -5, -9, -2}

In[563]:= Intersection[pom1, pom2]
Out[563]= {-9, -5, -3, -1}
```

Alebo ešte o kúsok lepšia optimalizácia - môžete kombinovať zadávané podmienky pomocou And - && , alebo pomocou Or - ||

```
In[564]:= Select[A // Flatten , Mod[#, 2] != 0 && # < 0 &]
Out[564]= {-1, -3, -5, -9}
```

- nahradte tretí riadok samými dvojkami

1. spôsob

```
In[565]:= A = {{-2, 3, 5}, {-1, -3, 8}, {1, -5, -9}, {-2, 2, 4}}
Out[565]= {{-2, 3, 5}, {-1, -3, 8}, {1, -5, -9}, {-2, 2, 4}}

In[566]:= A[[3]]
Out[566]= {1, -5, -9}

In[567]:= A[[3]] = {2, 2, 2}
Out[567]= {2, 2, 2}

In[568]:= A
Out[568]= {{-2, 3, 5}, {-1, -3, 8}, {2, 2, 2}, {-2, 2, 4}}
```

nahradte prvky poľa, ktoré ležia v treťom a štvrtom riadku a v druhom a treťom

stĺpci nulami

In[569]:= **A // MatrixForm**

Out[569]//MatrixForm=

$$\begin{pmatrix} -2 & 3 & 5 \\ -1 & -3 & 8 \\ 2 & 2 & 2 \\ -2 & 2 & 4 \end{pmatrix}$$

$$\begin{pmatrix} -2 & 3 & 5 \\ -1 & -3 & 8 \\ 2 & 2 & 2 \\ -2 & 2 & 4 \end{pmatrix}$$

In[570]:= **A[[3 ;; 4, 2 ;; 3]]**

Out[570]= **{{2, 2}, {2, 4}}**

{{2, 2}, {2, 4}}

toto sme vás nikdy neučili - dá sa to spraviť napríklad cez pattern

In[571]:= **ReplacePart[A, {3 | 4, 2 | 3} → 0]**

Out[571]= **{{-2, 3, 5}, {-1, -3, 8}, {2, 0, 0}, {-2, 0, 0}}**

{{-2, 3, 5}, {-1, -3, 8}, {1, 0, 0}, {-2, 0, 0}}

odstráňte štvrtý riadok matice

In[572]:= **Delete[A, 4]**

Out[572]= **{{-2, 3, 5}, {-1, -3, 8}, {2, 2, 2}}**

- pridajte na začiatok riadok, ktorý je tvorený prvkami {5, 7, 9}

In[573]:= **Prepend[A, {5, 7, 9}]**

Out[573]= **{{5, 7, 9}, {-2, 3, 5}, {-1, -3, 8}, {2, 2, 2}, {-2, 2, 4}}**