

Листе

Листе су сложени типови података и базирају се на осталим типовима података.

Сваки податак у листи јесте елемент листе и њему се може приступити преко његове позиције у листи.

Позиција елемента у листи је одређена његовим индексом.

Индекс је целобројна вредност и први елемент у листи има индекс 0.

Елементи листе се увек смештају у повезаним меморијским локацијама.

Листа може бити једнодимензионалана, дводимезионалана (матрица) или вишедимензионалана (коцка).

Код једнодимензионалне листе димензија је и дужина листе (листа је димензије n).

Код вишедимензионалних листа се каже да је листа димензија $m \times n \times z$.

Вишедимензионалне листе

Када листа за своје елементе има друге листе, таква листа се назива вишедимензионална листа:

```
podlista_1 = [[lista_A_1], [lista_A_2], ...[lista_A_n]]
```

```
podlista_2 = [[lista_B_1], [lista_B_2], ...[lista_B_n]]
```

...

```
ime_visedimenzionalne_liste = [[podlista_1], [podlista_2], ...[podlista_n]]
```

Од вишедимензионалних листи најчешће се користе дводимензионалне листе или матрице.

Дводимензионалне листе - матрице

Број листа у матрици представља број редова(врста).

Број појединачних елемената у листи матрице је број колона.

Иницијализација матрица

```
A = [[1, 2, 3], [4, 5, 6]] #matrica A je inicijalizovala dve vrste po tri kolone elemenata
```

Пре било какве модификације матрице, матрица се мора иницијализовати, тј. матрица мора бити попуњена са неким елементима.

Могуће је прво доделити матрици само један елемент а касније додати и остале.

0103 Иницијализација матрице

```
A = [[0]]
```

```
print("Matrica A se sastoji od", A)
```

```
A = [[0], [0]]
```

```
print("Matrica A je sada", A)
```

У примеру је матрица A иницијализована као једнодимензионална листа са једним елементом 0.

Касније је постала дводимензионална листа (матрица), са два реда и једном колоном.

Сада је то матрица A реда 2 x 1.

Иницијализација матрица са истим елементима

Ако је познато унапред какве су димензије потребне матрице, најлакше је иницијализовати матрицу постављањем свих елемената на исте почетне вредности, најчешће на 0.

0104 Иницијализација матрице list comprehension

Иницијализовати матрицу A 3x3

```
A = [[0 for x in range(3)] for y in range(3)]
```

што је исто као да је иницијализована директним уписивањем потребног броја елемената по подлистама:

```
A = [[0, 0, 0], [0, 0, 0], [0, 0, 0]]
```

Изостављање појединих елемената матрице

Пајтон допушта да се не попуне сва места у матрици са елементма, и то само као последњи елементи у редовима.

0105 Матрица као листа

```
T = [[11, 12, 5], [15, 6, 10], [10, 8, 12, 5], [12, 15, 8, 6]]
print(T)
```

што даје: [[11, 12, 5], [15, 6, 10], [10, 8, 12, 5], [12, 15, 8, 6]]

У примеру се види да у прва два реда матрице има по три елемента (три колоне су попуњене) а у друга два су по четири елемента.

Пајтон допушта ову недоследност у реализацији матрице пошто су у они у сагласности са промењивости структуре типа података листа (оставља могућност да се накнадно унесу елементи).

Исто тако:

```
T = [[11, 12], [15, 6, 10], [10, 8, 12, 5], [12, 15, 8, 6, 1000]]
```

И ова матрица је синтаксно правилно иницијализована иако сваки ред матрице има неједнак број попуњених колона.