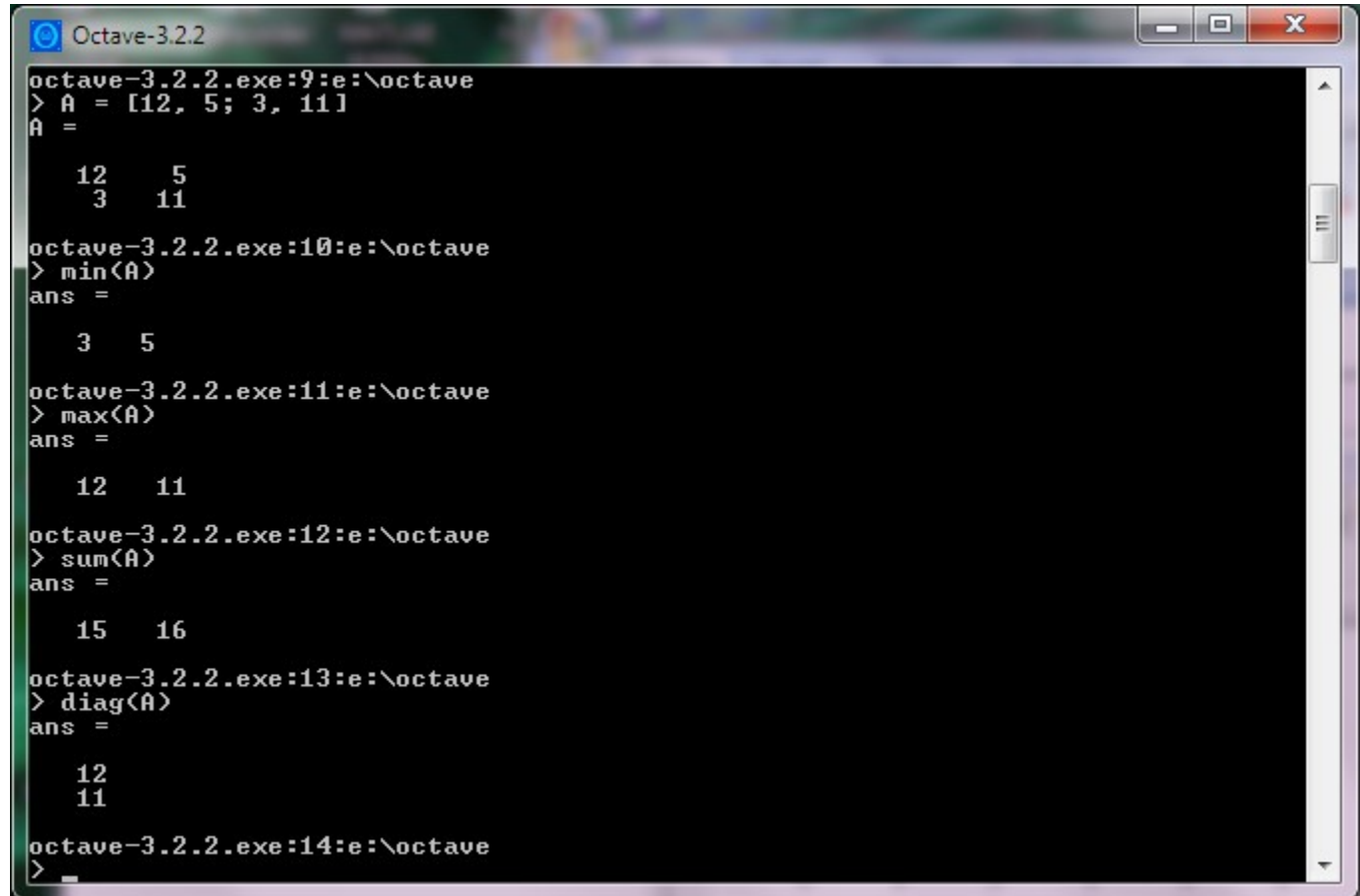


MATLAB

Magični kvadrat

Določene operacije

- ▶ $A = [12, 5; 3, 11]$
- ▶ `min(A)`
- ▶ `max(A)`
- ▶ `sum(A)`
- ▶ `diag(A)`



```
Octave-3.2.2
octave-3.2.2.exe:9:e:\octave
> A = [12, 5; 3, 11]
A =
    12    5
     3   11

octave-3.2.2.exe:10:e:\octave
> min(A)
ans =
     3     5

octave-3.2.2.exe:11:e:\octave
> max(A)
ans =
    12    11

octave-3.2.2.exe:12:e:\octave
> sum(A)
ans =
    15    16

octave-3.2.2.exe:13:e:\octave
> diag(A)
ans =
    12
    11

octave-3.2.2.exe:14:e:\octave
> _
```

"Rezanje" matrik

▶ Element

▶ $A(2, 3)$

▶ Podmatrika

▶ $A(2:4, 1:3)$

▶ Vrstica

▶ $A(2, :)$

▶ Stolpec

▶ $A(:, 4)$

```
>> matA

matA =

     1     2    23
     3     5     6
     6    12     3
     3     4     1
```

```
>> matA(4,2)
```

```
ans =
```

```
4
```

```
>> matA(2:4,2:3)
```

```
ans =
```

```
     5     6
    12     3
     4     1
```

```
>> matA(:,2:3)
```

```
ans =
```

```
     2    23
     5     6
    12     3
     4     1
```



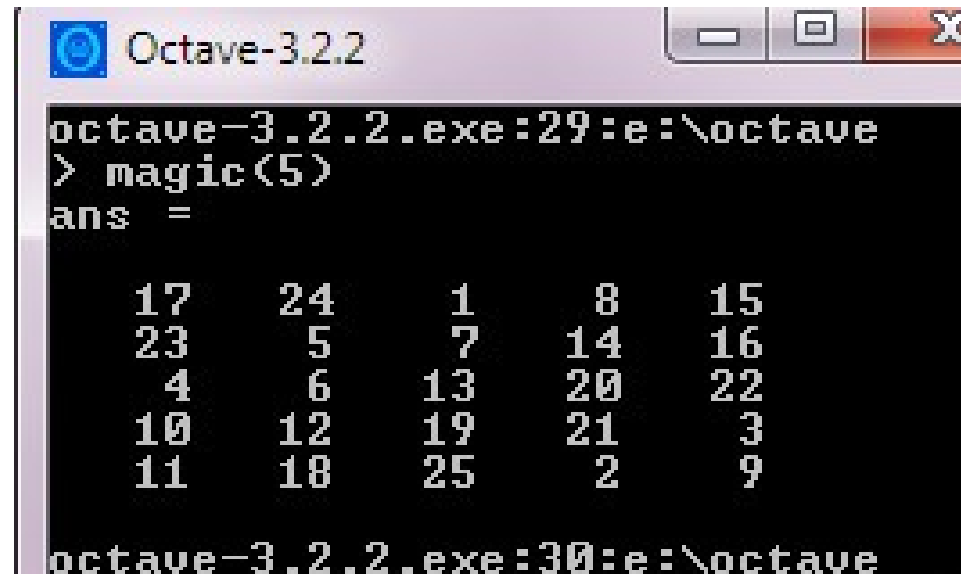
Komentiranje

- ▶ % začne komentar
- ▶ `matI = eye(5) % matI je identična matrika velikosti 5`



Naloga

- ▶ Sestavi magični kvadrat velikosti 5 x 5 in preveri, če je vse OK (diagonali, stolpci in vrstice seštejejo v isto vsoto)
- ▶ `help magic`
- ▶ `magKv = magic(5)`
- ▶ Hura – le še kontrole
 - ▶ `sum(magKv)`
 - ▶ Vsote po stolpcih
 - ▶ Kaj pa po vrsticah?
 - ▶ `sum(magKv')`
 - ▶ Preverimo glavno diagonalo
 - ▶ `sum(diag(magKv))`
 - ▶ Kaj pa obratna?
 - ▶ Pomoč ...
 - ▶ `rot90`
 - ▶ `sum(diag(rot90(magKv)))`



```
Octave-3.2.2
octave-3.2.2.exe:29:e:\octave
> magic(5)
ans =

    17    24     1     8    15
    23     5     7    14    16
     4     6    13    20    22
    10    12    19    21     3
    11    18    25     2     9

octave-3.2.2.exe:30:e:\octave
```

Še drugačen način

- ▶ Ali gre brez transponiranja in rotacij?
- ▶ Kaj počne
 - ▶ `sum(A, 1)`
 - ▶ `sum(A,2)`
- ▶ Kako pa dobiti nasprotno diagonalo?
 - ▶ Neposredno ne bo šlo ...
 - ▶ `rot90` je morda najlažje!
 - ▶ Kaj pa `flipdim(A,1)` ali `flipdim(A,2)`
 - ▶ `B = flipdim(A,dim)` returns A with dimension dim flipped.
 - ▶ When the value of dim is 1, the array is flipped row-wise down. When dim is 2, the array is flipped columnwise left to right. `flipdim(A,1)` is the same as `flipud(A)`, and `flipdim(A,2)` is the same as `fliplr(A)`

