

7. Matlab - formátovaný výstup v Matlabe

Blaho Michal · MATLAB/Comsol

26.06.2009



V predchádzajúcej časti nášho seriálu o Matlabe sme si popisovali znaky a textové reťazce. Výstup výsledkov môžeme formátovať v príkazovom riadku (Command Window) pomocou niekoľkých typov formátov. Medzivýsledok môžeme naformátovať aj pomocou príkazov, ktoré vrátia výsledok ako textový reťazec.

Formát v príkazovom riadku

Ako už bolo spomenuté, v príkazovom riadku (Command Window) dokážeme zmeniť formát výstupu. Treba dbať nato, že meníme len formát zobrazovania výsledku a nie ako ho Matlab ukladá. Na zmenu formátu výsledku slúži príkaz ***format***. Príkaz má nasledujúci zápis:

```
>>format typ
>>format('typ');
```

V nasledujúcej tabuľke sa nachádzajú rôzne druhy formátov pričom ich výstup si ukážeme na Ludolfolom čísle ***pi***. Tabuľka obsahuje formáty pre čísla s pohyblivou čiarkou.

Formát	Popis	Výsledok
short	zmenšené číslo so 4 číslami za desatinnou čiarkou	3.1416
long	zmenšené číslo so 14 alebo 15 číslami za desatinnou čiarkou pre typ double a so 7 číslami pre typ single	3.141592653589793
short e	formát čísla s pohyblivou čiarkou so 4 číslami za desatinnou čiarkou	3.1416e+00
long e	formát čísla s pohyblivou čiarkou so 14 alebo 15 číslami za desatinnou čiarkou pre typ double a so 7 číslami pre typ single	3.141592653589793e+00
short g	lepšie z čísel v pevnej rádovej čiarke alebo v pohyblivej rádovej čiarke so 4 číslami za desatinnou čiarkou	3.1416

long g	lepšie z čísel v pevnej rádovej čiarky alebo v pohyblivej rádovej čiarky so 14 alebo 15 číslami za desatinnou čiarkou pre typ double a so 7 číslami pre typ single	3.14159265358979
short eng	číslo v inžinierskom formáte s presne 4 číslami za desatinnou čiarkou a s 3 číslami v mocnine	3.1416e+000
long eng	číslo v inžinierskom formáte s presne 16 číslami za desatinnou čiarkou a s 3 číslami v mocnine	3.14159265358979e+000

Tab.1 Formát pre čísla s pohyblivou čiarkou

Ďalšie formáty v nasledujúcej tabuľke slúžia pre všetky číselné typy.

Formát	Popis	Výsledok
+	vracia znamienko čísla	+, -, prázdne miesto
bank	menový formát, 2 čísla za desatinnou čiarkou (pozn. pri dlhých čísla môže vracať zlý výsledok)	3.14
hex	hexadecimálna (šestnástková) reprezentácia binárneho čísla s dvojitou presnosťou za desatinnou čiarkou.	400921fb54442d18
rat	číslo vo formáte zlomku pre malé celočísla	355/113

Tab.2 Ďalšie formáty pre číselné typy

Zvyšné formáty upravujú medzery vo výsledku. Pred použitím sme zadali formát short.

Formát	Popis	Výsledok
compact	odstraňuje medzery z príkazového riadku, aby sa zmestilo viac výsledkov	>> pi ans = 3.1416
loose	pridáva medzi výsledok medzery, aby sa výsledok dal čítať ľahšie	>> pi ans = 3.1416

Tab.3 Formátovanie medzier vo výsledku

Formát v príkazovom riadku si môžeme zmeniť aj pomocou horného menu vo File > Preferences > Command Window. Pre získanie aktuálne nastaveného formátu môžeme použiť príkaz **get** s parametrom **Format** a na zistenie zobrazovania medzier parameter **FormatSpacing** ako je to v nasledujúcom príklade.

```
>>format long
>>format compact
>>get(0, 'Format');
ans = long
>>get(0, 'FormatSpacing').
ans = compact
```

Formátovanie textového reťazca

Matlab poskytuje niekoľko funkcií, ktoré dokážu poskladať textový reťazec z obyčajného textu a iných dát (napríklad čísla, biele znaky, atď.). Formátovanie je podobné formátovaniu vo funkcii **printf** v programovacom jazyku C. V nasledujúcej tabuľke sú funkcie, ktoré dokážu formátovať reťazec.

Funkcia	Popis
sprintf	zapisuje formátovaný text do textovej premennej
fprintf	zapisuje formátovaný text do súboru alebo Command Window
warning	zobrazuje formátovaný text ako upozornenie
error	zobrazuje formátovaný text ako chybu a skončí vykonávanie programu
assert	vyhlási chybu pri nedodržaní podmienky

Tab.4 Formátovanie textu vo funkciách

Hlavnou časťou argumentov funkcií je formátovaný text a hodnoty, ktoré sú doplnené do textu. Príkazy majú teda nasledujúcu štruktúru

>>príkaz(param,formáovaný text,hodnota1,...,hodnotaN)

Formátovaný text je uzatvorený do apostrofov. Formátovaný text sa skladá z niekoľkých častí: **obyčajného textu**, **špeciálnych znakov** a **formátovacích operátorov**. Jednotlivé časti si detailne popíšeme. **Obyčajný text** sú znakové reťazce, ktoré bežne zadávame. **Špeciálne znaky** sú znaky textového reťazca, ktoré sa nedajú zadať priamo (pretože majú aj iný význam). Preto ich musíme zadávať pomocou nasledujúcich znakov v ďalšej tabuľke.

Popis znaku	Zadáваме ako
apostrof	"
znak percenta	%%
lomítko	\\
backspace	\b
posun formulára	\f
nový riadok	\n
carriage return	\r
tabulátor	\t
šestnástkové číslo N	\xN
osmičkové číslo N	\N

Tab.5 Špeciálne znaky

Formátovacie operátory hovoria ako má Matlab naformátovať dátové argumenty a kde ich má zapísať do textu. **Formátovací oprátor** začína znakom percenta a pokračuje ďalšími časťami, ktoré si popíšeme. Zapisujú sa v tomto poradí:

- **Identifikátor** - poradové číslo dátového argumentu
- **Prepínač** - kontroluje zarovnanie a vloženie znaku plus a mínus

- **Dĺžka poľa** - minimálny počet znakov na zobrazenie
- **Presnosť** - počet znakov za desatinnou čiarkou
- **Podtyp** - špecifikuje neštandardné typy
- **Znak konverzie** - špecifikuje notáciu výstupu

Všetky časti si popíšeme a ukážeme na príkladoch. **Znak konverzie** špecifikuje notáciu výstupu. Je to posledný znak oprátora a je jediný, ktorý musí byť zadaný. V nasledujúcej tabuľke sú všetky znaky na konverziu.

Znak	Popis
c	jeden znak
d	celé číslo (so znamienkom)
e	číslo v exponenciálnom tvare
E	číslo v exponenciálnom tvare s veľkým E v exponente
f	číslo s pevnou rádovou čiarkou
g	kompaktnejšie ako %e alebo %f
G	ako %g, ale s veľkým E v exponente
o	osmičkové číslo (bez znamienka)
s	textový reťazec
u	celé číslo (bez znamienka)
x	šestnástkové číslo (malé písmená a-f)
X	šestnástkové číslo (veľké písmená A-F)

Tab.6 Znaky konverzie

Nasledujúci príklad ukazuje zapísanie čísla 46 v rôznych dátových typoch.

```
>> sprintf('Číslo 46: %d, %f, %e, %X', 46, 46, 46, 46);
Číslo 46: 46, 46.000000, 4.600000e+01, 2E.
```

Podtyp je znak, ktorý sa môže dať pred znakom konverzie. Podporujú ho len niektoré typy ako %o, %u, %x a %X.

Znak	Popis
b	základný typ v jazyku C je skôr double ako celé číslo bez znamienka
t	základný typ v jazyku C je skôr float ako celé číslo bez znamienka

Tab.7 Podtyp

Podtyp teda hovorí Matlabu, že má číslo reprezentovať skôr ako číslo s dvojnásobnou presnosťou ako celé číslo bez znamienka. **Presnosť** je nezáporné celé číslo, ktoré hovorí koľko čísiel sa má nachádzať za desatinnou čiarkou. Pred presnosť treba dať vždy bodku.

```
>> sprintf('Presnosť: %.2g, %f, %.2f.', 46*pi, 46*pi, 46*pi)
Presnosť: 1.4e+02, 144.513262, 144.51.
```

Dĺžka poľa určuje koľko bude mať dátový argument minimálny počet znakov, preto to musí byť kladné celé číslo. Ak je číslo väčšie ako počet znakov čísla, ktoré chceme zobrazíť, zvyšné polia sa vyplnia medzerami.

```
>> sprintf(' |%e| %15e| %f| %15f| ', 46*pi, 46*pi, 46*pi, 46*pi)
|1e+02| 1.445133e+02|144.513262| 144.513262|
```

Prepínačom môžeme zarovnať výstup, zobrazíť znamienko alebo vyplniť medzery nulami. V nasledujúcej tabuľke sú vypísané prepínače.

Znak	Popis
znak mínus (-)	zarovnanie doľava
znak plus (+)	zobrazenie znamienka
0	doplňa nulu namiesto medzier pred dátami

Tab.8 Prepínače

Znaky môžeme medzi sebou aj kombinovať. Nasledujúce príklady ozrejmiť ich použitie.

```
>> sprintf(' |%6.2f| % -6.2f| ', pi, pi)
| 3.14|3.14 |
>> sprintf(' | %+6.2f| % -+6.2f| ', pi, pi)
| +3.14|+3.14 |
>> sprintf(' | %06.2f| % -06.2f| ', pi, pi)
|003.14|3.14 |
```

Matlab dosadzuje dátové premenné do textu v poradí v akom sú zadané. Môžeme mu však sami povedať v akom poradí ich má dosadiť do textu. Poradie určuje číslo za znakom percenta, za ktorým nasleduje symbol dolára \$.

```
>> sprintf('%s %s %s', 'a', 'b', 'c')
a b c
>> sprintf('%3$s %2$s %1$s', 'a', 'b', 'c')
c b a
```

Matlab neumožňuje usporiadať iba časť dátových premenných. Ak by sme všetkým dátovým premenným nepriradili poradie vypísal by chybu. Pre formátovanie dĺžky a presnosti platí zopár pravidiel, ktoré si uvedieme.

- ak nie je daná presnosť je štandardne 6
- ak je presnosť (p) menšia ako počet čísel za desatinnou čiarkou zobrazí sa len p čísel a zvyšok je zaokrúhlený
- ak je presnosť (p) väčšia ako počet čísel za desatinnou čiarkou zobrazí sa len p čísel a zvyšok je doplnený nulami
- ak nie je daná dĺžka poľa je štandardne nastavená ako presnosť plus 1 plus počet číslic pred desatinnou čiarkou
- ak dĺžka poľa (w) je viac ako p+1 plus počet číslic pred desatinnou čiarkou je reťazec rozšírený o w-(n+p+1) medzier alebo núl v závislosti od prepínača.

Dĺžku poľa a presnosť môžeme určiť aj vstupnými argumentami tak, že namiesto čísla dĺžky a presnosti dáme symbol hviezdičky *. Viac nám bude zrejmé z príkladu.

```
>> sprintf('Moje pi: %0*.*f',8,4,pi)
Moje pi: 003.1416
```

Dnes sme si teda ukázali ako dokážeme Matlabu povedať, aby nám zobrazoval výsledok v požadovanom formáte a ako si sami naformátujeme textový reťazec na základe dátových vstupov. Formátovaný text potom využijeme v ďalších funkciách, ktoré ako vstup požadujú textový reťazec. V nasledujúcej časti si popíšeme dôležitý dátový typ a tým je štruktúra.
