

10

BASH script programozás
case, while, függvény,
exit állapot, szűrés,
környezeti változók

Microsoft Build 2016

March 30 - April 1, 2016

Coverage

- Microsoft
- bash
- msbuild2016

Microsoft is bringing the Bash shell to Windows 10

Posted 17 hours ago by [Frederic Lardinois \(@fredericl\)](#)

27.4k SHARES

Popular Posts

- Oculus Rift Review**
2 days ago
- This War On Math Is Still Bullshit**
4 days ago
- CockroachDB just raised \$20 million from Benchmark, Index, and GV**
a day ago
- Sorry Sunrise, I'm switching to Fantastical**
a day ago
- Oculus Rift is amazing, but you probably shouldn't buy one**



Here is an announcement from Microsoft Build you probably didn't see coming: Microsoft today announced that it is bringing the [GNU project's Bash shell](#) to Windows. Bash (Bourne Again SHell) has long been a standard on OS X and many Linux distribution

CrunchBase

Microsoft

FOUNDED
1974

OVERVIEW
Microsoft is an American multinational corporation that develops, manufactures, licenses, supports and sells computer software, consumer electronics and personal computers and services. Its best known software products are the [Microsoft Windows line of operating systems](/product/windows), [Microsoft Office office suite](/product/microsoft-office), and [Internet Explorer](http://windows.microsoft.com/en-us/internet-explorer/download-ie) ...

LOCATION
Redmond, WA

CATEGORIES
Software, Video Games

WEBSITE
<http://www.microsoft.com>

[Full profile for Microsoft](#)

Logikai műveletek

case \$VÁLTOZÓ in

minta1)

 PARANCS(OK);;

minta2|minta3)

 PARANCS(OK);;

*)

 PARANCS(OK)

esac

while, until

while [FELTÉTEL]; do

 PARANCS(OK)

done

- vagy másképpen

while ((FELTÉTEL)); do

– PARANCS(OK)

done

- ehhez hasonlóan létezik **until** vezérlési szerkezet (amíg nem teljesül a feltétel)

függvények, függvényhívás

```
function FUGGVENYNEV() {  
    PARANCS(OK)  
}
```

```
FUGGVENYNEV PARAM1 PARAM2 ...  
PARAMN
```

if - kiegészítés

- if [kapcsolós feltétel]; then ✓
- if ((matematikai kifejezéses feltétel)); then ✓
- if PARANCS; then
 - a PARANCS kilépési állapota alapján lesz igaz vagy hamis →
 - hiba nélküli futás: true
 - hiba esetén pedig: false

Kilépési állapot (exit state)

- Általában a kilépési állapot:
 - exit 0 ha minden OK (if esetén true)
 - exit 1 valamilyen hiba történt (if esetén false)
- Az exit status 0 – 255 vehet fel értéket, tehát van lehetőség saját hiba kilépési státusz használatára is.
- PARANCS
if ((\$? == 0)); then

#Ahol a \$? az előző PARANCS exit állapotára hivatkozó változó.

Feladat

- hozzunk létre könyvtárat (név paraméterről)
 - ha sikeres, akkor írjuk ki, hogy SUCCESS
 - ha már létezik a fájl, vagy akkor pedig azt, hogy ERROR
 - a többi hibakimenetre írt tartalmat nyeljük el
 - használjunk függvényhívást
 - írassuk ki a függvényen belül a második paramétert

Szűrő

- grep – csak a mintához illeszkedő sorokat írja fájlból
- kapcsolók:
 - n: kiírja az illeszkedő sor sorszámát
 - c: megszámlálja azokat a sorokat, ahol illeszkedést talált
- Feladat töltsük le a gyakorlat honlapjának a html kódját és keressük meg benne a „zh” kifejezést.

Környezeti változók

- környezet:
 - név – érték párok halmaza
- (globális) shell szintű változók
- env, printenv: környezeti változók listázása
- export VALTOZO_NEV vagy export VALTOZO_NEV=érték:
 - környezeti változó létrehozása
- unset VALTOZO_NEV: környezeti változó feloldása

Feladat:

- hozzunk létre egy környezeti változót
- egy másik terminálban nézzük meg létezik-e
- egy alshellben is nézzük meg, mivan ha itt megszüntetjük