

Operációs rendszerek alapjai

V. gyakorlat

Random, tesztelés, hibakeresés, gyakorlás

\$RANDOM

- a bash random változója
- minden alkalommal, amikor használjuk egy új random szám generálódik
 - 16 bites egész
 - 0 és 32767 közötti szám
 - *echo \$RANDOM*
 - adott intervallumba eső véletlen szám generálása:
 - modulo és offset használatával
 - $\$((\$RANDOM \% i)) - [0 , i [$
 - $\$((\$RANDOM \% i + n)) - [n , i + n [$

bash -v

- A parancsokat kiírja a futtatása előtt, (de a változók értékét nem helyettesíti be, illetve a ciklusokat nem írja ki folyamatában)
- bekapcsolható több módon is:
 - A scriptben:
 - `#!/bin/bash -v` - az egész scriptre vonatkozik
 - `set -v` - az utána lévő részekre vonatkozik
 - `set +v` - felfüggeszti a -v funkciót
 - Futtatáskor
 - `bash -v script.sh` - szintén a teljes scriptre vonatkozik

bash -x

- a változók értékét megjeleníti, a ciklusokat lépésenként írja ki
- azonos módon kapcsolható be, mint a -v:
 - A scriptben:
 - `#!/bin/bash -x` - az egész scriptre vonatkozik
 - `set -x` - az utána lévő részekre vonatkozik
 - `set +x` - felfüggeszti a -x funkciót
 - Futtatáskor
 - `bash -x script.sh` - szintén a teljes scriptre vonatkozik
- Kombinálható a v-vel:
 - `bash -xv`

bash -n

- A parancsok nem hajtódnak végre, csak a szintaxis lesz ellenőrizve
- hasonlóan kapcsolható:
 - A scriptben:
 - `#!/bin/bash -n` - az egész scriptet csak ellenőrzi
 - `set -n` - az utána lévő részek már nem hajtódnak végre, csak ellenőrizve lesznek
 - Futtatáskor
 - `bash -n script.sh` - szintén a teljes scriptre vonatkozik, csak szintaxis ellenőrzés fog történni

G05F01

- Készítsünk egy kő-papír-olló játékot
 - a felhasználó a paper–scissors–rock valamelyikét begépel, a gép is random választ ezek közül majd kiírja mit választott és azt is, hogy ki győzött (döntetlent esetén pedig a döntetlent)
 - a játék addig tart míg, a felhasználó exit utasítást nem ad
 - a játék végén írja ki a gép az összesített eredményt
 - felhasználó x győzelem, gép y győzelem, z döntetlen

G05F02

- Készítsünk palindrom ellenőrző programot!
 - a palindrom olyan szó vagy szókapcsolat, mely visszafelé olvasva is ugyanaz
 - a szóköz, a vessző és a mondatvégi írásjelek (.,?,!) illetve a kis és nagybetűk nem számítanak
 - a program kérjen be egy szöveget a felhasználótól
 - ha a beírt szöveg palindrom, akkor írja ki, hogy “Palindrom”, különben pedig azt, hogy “Ez nem palindrom!”

Példa palindromok: Óh, a kiséger lohol! Rég esik a hó?

Ma ráz az áram.

Was it a cat I saw?

Hint: `String.charAt(i)` - `String.length()` (a palindrom i-edik karaktere)

G05F03

- Készítsünk process számoló programot!
 - a program 2 másodperces időközönként számolja meg az összes process számát, összesen 5 alkalommal
 - amikor elkezd futni írja ki, hogy Busy és minden számolás után írjon utána egy pontot
 - végül a kapott 5 szám átlagát írja ki a képernyőre

G05F04

- Csináljunk egy programot ami minden bash parancsot végrehajt!
 - kérjen be egy parancsot a felhasználótól a terminálon keresztül
 - ha a felhasználó utasítása az exit volt, akkor kérdezzen rá, hogy biztosan ki akar-e lépni
 - ha a felhasználó igent mond, akkor lépjen ki (zárja be a terminált is)
 - különben kérje be a következő parancsot
 - minden más parancsot kérdés nélkül hajtson végre
 - új parancsot addig kérjen míg a felhasználó ki nem lép

Néhány jó tanács (a ZH előtt)

- Tab - automatikus parancs kiegészítés
- ↑ és ↓ nyilak segítségével barangolhatunk a korábbi parancsok között
- *man parancs* - megkapjuk a parancs leírását
- *apropos valami* - a parancsok leírásaiban keresi a valamit
- *chmod +x fájl* - futtathatóvá teszi a fájlt
- *./script* - az aktuális mappában lévő script futtatása
- *clear* - a terminált törli ((pontosabban csinál jó sok üres sort...))