

Számítógép Architektúrák

BSc

9. Gyak.

2023. 12. 06.

Készítette:

Takács Antal Levente BSc
Programtervező Informatikus
TB1EY5

Miskolc, 2023

1.feladat – Írjon ki 0-tól 10-ig a számokat,
for,
while i=0
until ciklussal is!

```
1   echo 'For:'
2
3   for i in {1..10}
4   do
5   echo $i
6   i=i-1
7   done
8
9   echo 'while:'
10
11  declare -i i=0
12  while [ $i -le 10 ]; do
13      echo $i
14      i=$((i + 1))
15  done
16
17  echo 'until:'
18
19  declare -i i=0
20  until [ $i -gt 10 ]; do
21      echo $i
22      i=$((i + 1))
23  done
```

```
tbley5@jerry:~/tbley5_szgyak/gyak9$ bash ciklus1.sh
1
2
3
4
5
6
7
8
9
10
tbley5@jerry:~/tbley5_szgyak/gyak9$
```

2.feladat – Generáljon egy véletlen számot és írja ki a nála kisebb négyzetszámokat!

```
1  #!/bin/bash
2
3  declare -ri n=$RANDOM
4
5  echo "A random szán: $n"
6
7  declare -i i=0
8  declare -i square_i=0
9  while [ $square_i -lt $n ]; do
10     echo $square_i
11     i=$((i + 1))
12     square_i=$((i * i))
13  done
```

tbley5@jerry:~/tbley5_szgyak/gyak9\$ bash ciklus2.sh
A random szán: 31810
0
1
4
9
16
25
36
49
64
81
100

3.feladat – Írjon egy shell scriptet, ami bemeneti paraméterként egyetlen pozitív számot vár (hibát ír, ha nem ezt kap).

A program feladata, hogy kiírja, hogy a kapott szám prím-e.

```
read input
if [[ $input ]] && [ $input -eq $input 2>/dev/null ]
then
    echo "$input egy szam"
    i=2

if [ $input -lt 2 ]
then
    echo "$input nem prim"
    exit
fi

while [ $i -lt $input ]
do
    if [ `expr $input % $i` -eq 0 ]
    then
        echo "$input nem prim"
        exit
    fi
    i=`expr $i + 1`
done

echo "$input prim szam"

else
    echo "$input nem egy szam"
fi
```

```
tbley5@jerry:~/tbley5_szgyak/gyak9$ bash ciklus3.sh
4
4 egy szam
4 nem prim
tbley5@jerry:~/tbley5_szgyak/gyak9$
```

4.feladat – Töltsön fel egy 10 elemű tömböt véletlen számokkal, majd menjen végig a számokon és növelje meg őket 1-gyel!

```
1  for ((i=0;i<=10;i++)); do
2  array[$i]=$((RANDOM % 100))
3  done
4
5  echo "Tomb: ${array[@]}"
6
7  for ((i=0;i<=10; i++)); do
8  array[$i]=$((array[$i]+1))
9  done
10
11 echo "1el novelt tomb: ${array[@]}"
```

```
tbley5@jerry:~/tbley5_szgyak/gyak9$ bash tomb1.sh
Tomb: 90 74 58 74 2 51 28 43 1 41 83
1el novelt tomb: 91 75 59 75 3 52 29 44 2 42 84
tbley5@jerry:~/tbley5_szgyak/gyak9$
```

5.feladat – Hozzon létre egy N elemű tömböt, ahol N-t read paranccsal kérje be, és ellenőrizze, hogy pozitív egész szám (hibával visszatérünk, ha nem az).

Töltse fel véletlen számokkal a tömböt, ahol a véletlen számok az [1-100] intervallumból kerülnek ki!

Ezután végezze el a következő műveleteket a tömbre:

- Minimum elem kiírása (hányadik elem és mi az értéke).
- Maximum elem kiírása (hányadik elem és mi az értéke).
- Írja ki az elemek összegét.
- Írja ki az elemek átlagát.

```
1  read -r num
2  for ((i=0;i<num;i++)); do
3  array[$i]=$((RANDOM % 100))
4  done
5
6  echo "tomb: ${array[@]}"
7
8  min=${array[0]}
9  max=${array[0]}
10 mindex=0
11 mxindex=0
12
13 for ((i=1;i<num;i++)); do
14 if ((array[i] < min)); then
15 min=${array[i]}
16 mindex=$i
17 fi
18
19 if ((array[i] > max)); then
20 max=${array[i]}
21 mxindex=$i
22 fi
23 done
24
25 echo "legkisebb: $min, hanyadik elem: $mindex"
26 echo "legnagyobb: $max, hanyadik elem: $mxindex"
27
28 declare -i sum=0
29 for i in $(seq 0 $((n - 1))); do
30     sum=$((sum + arr[i]))
31 done
32 echo "Összeg: $sum"
33
34 avg=$((sum / n))
35 echo "Átlag: $avg"
```

```
tbley5@jerry:~/tbley5_szgyak/gyak9$ bash tomb2.sh
Add meg a tomb meretet:
4
tomb: 90 87 61 25
legkisebb: 25, hanyadik elem: 3
legnagyobb: 90, hanyadik elem: 0
Összeg: 263
Átlag: 65
tbley5@jerry:~/tbley5_szgyak/gyak9$
```

6.feladat – Készítsen egy függvényt, ami összeadja az argumentumban megadott számokat. Ha nem kap argumentumot, akkor 0-t ír ki, egyébként az argumentumok összegét. Az argumentumokat nem szükséges ellenőrizni, hogy számok-e.

```
1  #!/bin/bash
2
3  function sum {
4      declare -i sum=0
5
6      for n; do
7          sum=$((sum + n))
8      done
9
10     echo $sum
11 }
12
13 sum "$@"
```

```
tbley5@jerry:~/tbley5_szgyak/gyak9$ bash fg1.sh 3 6 9
18
tbley5@jerry:~/tbley5_szgyak/gyak9$
```


7.feladat – Írjon függvényt, ami legenerál N darab véletlen számot egy [x-y] intervallumban. N, x és y értékeit paraméterekként kapja meg a függvény.

Ha bármelyik paramétert nem adják meg, akkor a default értékek legyenek a következők:

N=5, x=1, y=90.

Generáltassunk a függvénnyel 10 véletlen számot 800 és 900 között, majd 15 számot -10 és 10 között!

```
1  #!/bin/bash
2
3  function randgen {
4      declare -ri N=${1:-5}
5      declare -ri x=${2:-1}
6      declare -ri y=${3:-90}
7
8      for _ in $(seq 1 $N); do
9          echo $((RANDOM % (y - x + 1) + x))
10     done
11 }
12
13 randgen 10 800 900
14 randgen 15 -10 10
```

```
tb1ey5@jerry:~/tb1ey5_szgyak/gyak9$ bash fg2.sh
819
834
810
812
833
827
899
859
823
828
-7
-1
1
-2
7
-3
7
1
1
8
10
-4
-9
-9
1
tb1ey5@jerry:~/tb1ey5_szgyak/gyak9$
```