

יטודות מערכות פתוחות
פתרון תרגיל מס' 6

1. כתוב/כתבי תכנית Script ב-Bash בשם P6.1 שפועלת באופן הבא:

בהפעלת התכנית, מופיעה ההודעה הבאה:
Enter 3 parameters in format file-string-num:

והתכנית מחכה לקלט מהמשתמש שמכיל שם קובץ, מחרוזת ומספר i.
נניח למשל שהמשתמש מקליד: F1-def-3

נניח למשל שתוכן הקובץ F1 הוא:

```
12bdef def yydefgg x def y 5defdefdef
123cdefb 87adef312
6def7 8288a1bcc5 1deecc x defdef
ydef2 yyzzdef 4def x ww
z7de 1delf 6def
```

בשלב הזה התכנית מדפיסה לפלט את כל העמודות בקובץ שבהן המחרוזת מופיעה בדיוק i פעמים.

כל עמודה בפלט תופיע בשורה נפרדת כשבתחילת העמודה יופיע מספר העמודה (בפורמט שמתואר בדוגמה שבהמשך).

סדר העמודות הוא לפי מספרי העמודות, וסדר המילים בכל עמודה הוא לפי סדר הופעתן בקובץ.

בדוגמה הנ"ל, יתקבל הפלט:

```
2: def 87adef312 8288a1bcc5 yyzzdef 1delf
3: yydefgg 1deecc 4def 6def
5: def defdef ww
7: 5defdefdef
```

פתרון שאלה 1

```
1 echo -n "Enter 3 parameters in format file-string-num: "  
2 read cmd  
3 filename=$(echo $cmd | cut -d"-" -f1)  
4 str=$(echo $cmd | cut -d"-" -f2)  
5 num=$(echo $cmd | cut -d"-" -f3)  
6 >| temp  
7 max=0  
8 while read line; do  
9     cols=$(echo $line | wc -w)  
10    if [ $cols -gt $max ]; then  
11        max=$cols  
12    fi  
13    echo $line >> temp  
14 done < $filename  
15 for i in $(seq $max); do  
16    colstr=$(cut -d" " -f$i temp)  
17    if [ $(echo "$colstr" | grep -o $str | wc -w) -eq $num ]; then  
18        echo -n "$i: "  
19        echo $colstr  
20    fi  
21 done
```

להלן אותו פתרון בפורמט טקסט.

```
echo -n "Enter 3 parameters in format file-string-num: "  
read cmd  
filename=$(echo $cmd | cut -d"-" -f1)  
str=$(echo $cmd | cut -d"-" -f2)  
num=$(echo $cmd | cut -d"-" -f3)  
>| temp  
max=0  
while read line; do  
    cols=$(echo $line | wc -w)  
    if [ $cols -gt $max ]; then  
        max=$cols  
    fi  
    echo $line >> temp  
done < $filename  
for i in $(seq $max); do  
    colstr=$(cut -d" " -f$i temp)  
    if [ $(echo "$colstr" | grep -o $str | wc -w) -eq $num ]; then  
        echo -n "$i: "  
        echo $colstr  
    fi  
done
```

2. כתוב/כתבי תכנית Script ב-Bash בשם P6.2 שפועלת באופן הבא:

בהפעלת התכנית, מופיעה ההודעה הבאה:

Enter file name:

והתכנית מחכה לקלט מהמשתמש שמכיל שם קובץ. נניח למשל שהמשתמש מקליד: F2

בשלב הזה התכנית מדפיסה לפלט את כל השורות בקובץ שמקימות את כל התנאים הבאים.

המילה השנייה בשורה מכילה ספרות בלבד.

מספר הספרות במילה השלישית בשורה שווה למספר האותיות האנגליות (קטנות או גדולות).

המילה הלפני אחרונה בשורה לא מכילה ספרות.

המילה הרביעית בשורה היא בדיוק המילה four

על סדר השורות בפלט להיות לפי סדר הופעתן בקובץ.

לדוגמה, נניח שתוכן הקובץ F2 הוא:

```
ab 124444 #${ab}34! four 123
ab 124444 #${@}! foura
ab 124444 #${@}! four
ab 1 #${x@}! four
  ab 124444 #${@}! 1four
ab 124444 #${@}! four alb x
ab 124@ #${@}! four
  1 2 1a2b four zz 123
```

בדוגמה הנ"ל, יתקבל הפלט:

```
ab 124444 #${ab}34! four 123
ab 124444 #${@}! four
  1 2 1a2b four zz 123
```

שימו לב שיש לשמור בפלט על הרווחים שבשורות הקובץ כולל הרווחים שבתחילת כל שורה.

פתרון שאלה 2

```
1 echo -n "Enter file name: "  
2 read file_name  
3 counter=1  
4 while read line  
5 do  
6   wrd2=$(echo $line | cut -d" " -f2)  
7   wrd3=$(echo $line | cut -d" " -f3)  
8   wrd4=$(echo $line | cut -d" " -f4)  
9   n=$(echo $line | wc -w)  
10  wrd_before_last=$(echo $line | cut -d" " -f${n-1})  
11  
12  t1=$(echo $wrd2 | egrep -o [^0-9] | wc -l)  
13  t2=$(echo $wrd3 | egrep -o [0-9] | wc -l)  
14  t3=$(echo $wrd3 | egrep -o -i [a-z] | wc -l)  
15  t4=$(echo $wrd_before_last | egrep -c [0-9])  
16  
17  if [ $t1 -eq 0 ] && [ $t2 -eq $t3 ] && [ $t4 -eq 0 ] && [ "$wrd4" = "four" ]  
18  then  
19    head -$counter $file_name | tail -1  
20  fi  
21  ((counter++))  
22 done <$file_name
```

להלן אותו פתרון בפורמט טקסט.

```
echo -n "Enter file name: "  
read file_name  
counter=1  
while read line  
do  
  wrd2=$(echo $line | cut -d" " -f2)  
  wrd3=$(echo $line | cut -d" " -f3)  
  wrd4=$(echo $line | cut -d" " -f4)  
  n=$(echo $line | wc -w)  
  wrd_before_last=$(echo $line | cut -d" " -f${n-1})  
  
  t1=$(echo $wrd2 | egrep -o [^0-9] | wc -l)  
  t2=$(echo $wrd3 | egrep -o [0-9] | wc -l)  
  t3=$(echo $wrd3 | egrep -o -i [a-z] | wc -l)  
  t4=$(echo $wrd_before_last | egrep -c [0-9])  
  
  if [ $t1 -eq 0 ] && [ $t2 -eq $t3 ] && [ $t4 -eq 0 ] && [ "$wrd4" =  
"four" ]  
  then  
    head -$counter $file_name | tail -1  
  fi  
  ((counter++))  
done <$file_name
```

3. כתוב/כתבי תכנית Script ב-Bash בשם P6.3 שפועלת באופן הבא:

בהפעלת התכנית, מופיעה ההודעה הבאה:

Enter 4 parameter in format file-string1-string2-num:

והתכנית מחכה לקלט מהמשתמש שמכיל שם קובץ שתי מחרוזות ומספר i. נניח למשל שהמשתמש מקליד: F3-abcd-12345-3

בשלב הזה התכנית מדפיסה לפלט את כל המילים בקובץ שמקימות את התנאי הבא:

המילה מתחילה ב-i חזרות בדיוק של תווים שלא נמצאים במחרוזת 1 ולאחר מכן ההמשך הוא i חזרות בדיוק של תווים שנמצאים במחרוזת 2 ולאחר מכן ההמשך הוא בין תו אחד ל-i תווים שנמצאים במחרוזת 1 בשלב זה או שהמילה מסתיימת או ששוב מתקימת סדרת החוקים הנ"ל, דהיינו i חזרות של תווים שאינם במחרוזת 1, לאחריהם i חזרות של תווים שנמצאים במחרוזת 2 ולאחר מכן בין תו אחד ל-i תווים שנמצאים במחרוזת 1, בשלב זה או שהמילה מסתיימת או ששוב מתקימת סדרת החוקים הנ"ל, וכן הלאה ללא הגבלה.

על כל מילה בפלט להופיע בשורה נפרדת, וסדר המילים בפלט הוא לפי סדר לכסיקוגרפי עולה.

שימו לב שכאשר מבקשים i חזרות בדיוק של תווים שנמצאים במחרוזת זה לא בהכרח חזרה של אותו התו. לדוגמה אם המחרוזת היא abc אז במילה abaadac יש 7 חזרות של תווים שנמצאים במחרוזת.

לדוגמה, נניח שתוכן הקובץ F3 הוא:

```
123444a221555brrrr432bbb 11e666a 11f112bbbb
8$#231acd 8d$231acd www333awww333bwww333cwww333dd
ddd111abcaaa222 111abc111 111222c 111222c111222
```

בדוגמה הנ"ל, יתקבל הפלט:

```
111222c
123444a221555brrrr432bbb
8$#231acd
www333awww333bwww333cwww333dd
```

פתרון שאלה 3

```
1 echo -n "Enter 4 parameter in format file-string1-string2-num: "  
2 read a  
3  
4 file=$(echo $a | cut -d"-" -f1)  
5 str1=$(echo $a | cut -d"-" -f2)  
6 str2=$(echo $a | cut -d"-" -f3)  
7 num=$(echo $a | cut -d"-" -f4)  
8  
9 >| final  
10 for word in $(echo $(cat $file))  
11 do  
12     echo $word | egrep "^[^$str1]{$num}[$str2]{$num}[$str1]{1,$num})+$" >> final  
13 done  
14  
15 sort final
```

להלן אותו פתרון בפורמט טקסט.

```
echo -n "Enter 4 parameter in format file-string1-string2-  
num: "  
read a  
  
file=$(echo $a | cut -d"-" -f1)  
str1=$(echo $a | cut -d"-" -f2)  
str2=$(echo $a | cut -d"-" -f3)  
num=$(echo $a | cut -d"-" -f4)  
  
>| final  
for word in $(echo $(cat $file))  
do  
    echo $word | egrep "^[^$str1]{$num}[$str2]{$num}[$str1]  
{1,$num})+$" >> final  
done  
  
sort final
```

4. שאלה זו הופיעה במבחן מועד ג בשנת 2016.

הגדרה: עבור מטריצה A שמכילה מספרים (לא בהכרח ריבועית ולא בהכרח עם מספר שווה של עמודות בכל שורה) נגדיר את הקילוף של A כמטריצה שמתקבלת באופן הבא: עבור כל שורה שמכילה מספר מילים מקסימאלי מוזקים את המספר האחרון בשורה.

לדוגמה, עבור המטריצה A הבאה:

```
10 20 30
40
25 15
```

הקילוף של המטריצה A הוא המטריצה B הבאה:

```
10 20
40
25 15
```

הקילוף של המטריצה B הוא המטריצה C הבאה:

```
10
40
25
```

הקילוף של המטריצה C הוא מטריצה ריקה שלא מכילה אף איבר.

שימו לב שהקילוף של מטריצה שבכל השורות שלה מופיע מספר אחד בלבד הוא מטריצה ריקה. לדוגמה עבור המטריצה D הבאה:

```
30
40
60
```

הקילוף של המטריצה D הוא מטריצה ריקה.

כתוב/כתבי תכנית Script ב-Bash בשם P6.4 (אין להשתמש בפקודות awk ו-sed בשאלה זו) שפועלת באופן הבא:

בהפעלת התכנית, מופיעה ההודעה הבאה:

```
Enter list of files in format file1-file2-file3...:
```

והתכנית מחכה לקלט מהמשתמש שמכיל רשימת שמות קבצים שמכילים מטריצות של מספרים. ניתן להניח שבין המספרים ישנו רווח אחד בלבד ושארין רווחים בהתחלה ובסוף כל שורה. נניח למשל שהמשתמש מקליד: G1-G2

ונניח למשל שתוכן הקובץ G1 הוא:

```
20 4 7
10
3 50 12 10
```

ותוכן הקובץ G2 הוא:

```
80 100 200
```

בשלב הזה התכנית מדפיסה עבור כל קובץ ברשימת הקבצים:

שורה שמכילה את שם הקובץ, ולאחריה
שורה ריקה, ולאחריה
המטריצה עצמה, ולאחריה
שורה ריקה, ולאחריה
המטריצה שמתקבלת כתוצאה מהקילוף של המטריצה ולאחריה
שורה ריקה, ולאחריה
המטריצה שמתקבלת מהקילוף של הקילוף של המטריצה, ולאחריה
שורה ריקה,
וכן הלאה, עד שמגיעים למטריצה ריקה (מטריצה ריקה לא מודפסת).

על סדר המטריצות בפלט להיות לפי סדר הפרמטרים לתכנית.

בדוגמה הנ"ל, יתקבל הפלט:

G1

20 4 7
10
3 50 12 10

20 4 7
10
3 50 12

20 4
10
3 50

20
10
3

G2

80 100 200

80 100

80

פתרון שאלה 4

```
1 echo -n "Enter list of files in format file1-file2-file3...: "  
2 read files  
3  
4 for f in $(echo $files| tr "-" " ") ; do  
5     max=0  
6     >|temp4  
7     while read line; do  
8         wcnt=$(echo $line | wc -w)  
9         if [ $wcnt -gt $max ]; then  
10            max=$wcnt  
11        fi  
12        echo $line >> temp4  
13    done < $f  
14  
15    echo $f  
16  
17    for i in $(seq $max -1 1); do  
18        echo  
19        cut -d" " -f1-$i temp4  
20    done  
21    echo  
22 done
```

להלן אותו פתרון בפורמט טקסט.

```
echo -n "Enter list of files in format file1-file2-file3...: "  
read files  
  
for f in $(echo $files| tr "-" " ") ; do  
    max=0  
    >|temp4  
    while read line; do  
        wcnt=$(echo $line | wc -w)  
        if [ $wcnt -gt $max ]; then  
            max=$wcnt  
        fi  
        echo $line >> temp4  
    done < $f  
  
    echo $f  
  
    for i in $(seq $max -1 1); do  
        echo  
        cut -d" " -f1-$i temp4  
    done  
    echo  
done
```