

21.1.2011

**פתרון מבחן מועד א'**  
**יסודות מערכות פתוחות**  
**סמסטר חורף, תשע"א**

- משך המבחן: שלוש וחצי שעות.
- יש לענות על כל השאלות.
- מותר השימוש בחומר עזר כלשהו, פרט למחשבים, (מחשבוניס מותר).
- יש להקפיד על כתיבה ברורה ומסודרת של התשובות.

**1. (15 נקודות)**  
**חלק א (5 נקודות)**

הסעיף הזה לא בחומר של סמסטר חורף תשע"ג (כי לא למדנו expect) לכן לא כתבתי את הפתרון.

**חלק ב (10 נקודות)**

מה יתקבל בפלט (דהינו יוצג על המסך) לאחר הפעלת קטע הקוד הבא:

```
echo "output 1:"
echo -e "c3dc\ndcade\nccdc\nccdde" | egrep "c.d[^abc]"
```

```
echo "output 2:"
echo "abc def shalom" >| F1
echo " abc def shalom" >> F1
echo " abc def shalom123" >> F1
echo " abc def ghi shalom" >> F1
echo " abc def ghi shalom 456" >> F1
cat F1 | egrep "([ ]+[^ ]+){2}[ ]+shalom([ ]|$)"
```

```
echo "output 3:"
x=123456123456
echo ${x//12/ab}
echo ${x//[12]/cd}
echo ${x//[12][34]/e}
```

```
echo "output 4:"
A=(2 4 6 8 10 12 14 16 18 20)
echo ${A[@]:5:7}
s=0
for i in ${A[@]:4:3}
do
    s=$((s+$i))
done
```

```
echo s=$s
```

```
echo "output 5:"
```

```
echo "1 2 2 3 2 3 3 3" | tr " " "\n" | sort | uniq -c | sort -nr +0 -1
```

הפלט יהיה:

output 1:

```
dcade
```

```
ccddde
```

output 2:

```
abc def shalom
```

```
abc def ghi shalom
```

```
abc def ghi shalom 456
```

output 3:

```
ab3456ab3456
```

```
cdcd3456cdcd3456
```

```
1e4561e456
```

output 4:

```
12 14 16 18 20
```

```
s=6
```

output 5:

```
4 3
```

```
3 2
```

```
1 1
```

התשובה של output5 לא בחומר של השנה (כי לא למדנו sort כזה).

אין צורך לרשום איך התקבל הפלט מספיקה תשובה סופית.

## 2. (10 נקודות)

מילה הינה רצף תווים ללא תווי רווח וללא תווי סוף שורה.  
כתוב תוכנית Script ב- sed בשם P2 שמקבלת כפרמטרים שם קובץ (בהמשך נקרא לו קובץ 1) .  
התוכנית מדפיסה לפלט את מספר השורות בקובץ שמכילות לפחות שתי מילים.

על המבנה של התוכנית P2 להיות כדלהלן:

שורה ראשונה כמו בכל קובץ script ב- bash מכילה:

```
#!/bin/bash
```

לאחר מכן שורה אחת או יותר של פקודות sed לדוגמה

```
sed s/dog/cat/ $1
```

(אין להשתמש בפקודות שאינן של sed).  
מותר לכתוב לקובץ ביניים כמו למשל:

```
sed s/dog/cat/ $1 >| tmp
```

מותר גם לקרוא מקובץ ביניים כמו למשל:

```
sed s/dog/cat/ < tmp
```

אסור להשתמש ב- pipeline.

לדוגמה, נניח שתוכן הקובץ F1 הוא:

```
Helloabcd hi 123  
if a equals b  
you  
    one  
three agbcb four zzz
```

לאחר הפעלת התוכנית ע"י הפקודה:

```
>P2 F1
```

יתקבל הפלט:

```
3
```

פתרון שאלה 2

```
#!/bin/bash
```

```
sed '/^[ ]*[ ]+[ ]+[ ]+!/d' $1 >| tmp
```

```
sed -n '$=' tmp
```

### 3. (20 נקודות) הגדרה:

את מספר המילים שהופיעו לפנייה בשורה ועוד אחד. לדוגמא, האינדקס של המילה הראשונה בשורה הוא 1, האינדקס של המילה השנייה בשורה הוא 2, וכן הלאה. כתוב/כתבי תוכנית ב-awk בשם P3 שמקבלת כפרמטרים שם קובץ (בהמשך נקרא לו קובץ1) ולאחריו רשימה (באורך לא מוגבל) של מספרים (בהמשך נקרא לה רשימה1). התוכנית מדפיסה לפלט את השורות של קובץ1 לאחר צמצום המילים שהאינדקס שלהן אינו מופיע ברשימה1. בין המילים שבאותה השורה בפלט יודפס תו רווח יחיד. סדר המילים בכל שורה בפלט חייב להיות לפי סדר הופעתן בשורה שבקובץ (ולא לפי סדר המספרים שברשימה1).

**הערה:** אם מספר שמופיע ברשימה1 גדול יותר ממספר המילים בשורה, אזי המילה המתאימה לאינדקס זה אינה קימת והיא לא תופיע בפלט. יש לדאוג שבמקרה זה לא תודפס מחרוזת ריקה עבור מספר זה, על מנת שלא יופיע יותר מרווח אחד בין המילים בפלט.

לדוגמא, נניח שתוכן הקובץ F1 הוא:

```
ab 1234 777 def
hello hi bye shalom
xyz ww hh zz cabc kk aa
```

לאחר הפעלת התוכנית ע"י הפקודה:

```
>awk -f P3 F1 3 1 8 6 4
```

יתקבל הפלט:

```
ab 777 def
hello bye shalom
xyz hh zz kk
```

פתרון שאלה 3

```
BEGIN {
  for (x=2; x < length(ARGV); x++) {
    B[ARGV[x]] = 1
    ARGV[x] = "" # we put null string in ARGV[x] so that
                 # it will not be considered as a file name
                 # after the begin stage
  }
}
{
  s=""
  for (i=1; i<= NF; i++) {
    if (B[i] == 1) {
      s = s " " $i
    }
  }
  s=substr(s,2)
  print s
}
```

#### 4. (30 נקודות) בשונה

בשאלה זו נשתמש במונח **אינדקס** של מילה בשורה כפי שהוגדר בשאלה 3.

כתוב/כתבי תוכנית ב-Bash (דהינו קובץ Script) בשם P4 שמקבלת כפרמטרים רשימת מספרים באורך לא מוגבל (בהמשך נקרא לה רשימה1), ולאחר מכן המחזורת FILES ולאחר מכן רשימת שמות קבצים באורך לא מוגבל (בהמשך נקרא לה רשימה2). התוכנית מדפיסה לפלט עבור כל קובץ ששמו מופיע ברשימה2 שורה נפרדת ובה שם הקובץ ולאחר מכן את שורות של הקובץ כאשר בכל שורה מודפסות רק המילים שהאינדקס שלהן מופיע ברשימה1. בין המילים שבאותה השורה בפלט יודפס תו רווח יחיד. **בשונה** משאלה 3, על סדר המילים בכל שורה בפלט להיות לפי רשימה1 (ולא לפי סדר הופעתן בשורה בקובץ המתאים).

**הערה:** אם מספר שמופיע ברשימה1 גדול יותר ממספר המילים בשורה, אזי המילה המתאימה לאינדקס זה אינה קימת והיא לא תופיע בפלט. יש לדאוג שבמקרה זה לא תודפס מחזורת ריקה עבור מספר זה, על מנת שלא יופיע יותר מרווח אחד בין המילים בפלט.

לדוגמה, נניח שתוכן הקובץ F1 הוא:

```
one two three four five six
hello hi bye shalom
111 222 333 444 555 666 777
```

ונניח שתוכן הקובץ F2 הוא:

```
dog cat cow
abc def ghi jkl mnop
```

לאחר הפעלת התוכנית ע"י הפקודה:

```
>P4 3 8 9 1 7 5 FILES F1 F2
```

יתקבל הפלט:

```
F1
three one five
bye hello
333 111 777 555
F2
cow dog
ghi abc mnop
```

```
#!/bin/bash
index_list=""
for i in "$@"
do
  if [ $i = "FILES" ]
  then
    shift
    break
  fi
  index_list="$index_list $i"
  shift
done
for file in "$@"
do
  echo "$file"
  while read line
  do
    s=""
    for ind in $index_list
    do
      word=$(echo $line | cut -d " " -f$ind)
      s="$s $word"
    done
    echo $s
  done<"$file"
done
```

## 5. (25 נקודות) הגדרה: גודל

הגדרה: גודל קובץ שווה למספר הכולל של התווים בקובץ (כולל תווי רווח וסוף שורה). כתוב/כתבי תוכנית ב-Bash (דהינו קובץ Script) בשם P5 שמקבלת כפרמטרים שם תיקיה (בהמשך נקרא לה תיקיה1) ולאחריה רשימת מחרוזות (בהמשך נקרא לה רשימה1). התוכנית מדפיסה לפלט שורה אחת עבור כל מחרוזת שברשימה1 שמכילה את המחרוזת, לאחריה תו רווח יחיד ולאחריה מספר שמציין את גודלו של הקובץ הקטן ביותר מבין הקבצים ששםם (הסופי) שווה למחרוזת והם מופיעים בתיקיה1 בעומק כלשהו.

על סדר השורות בפלט להיות לפי סדר הופעת המחרוזות ברשימה1.

לדוגמא, נניח שתיקיה d מכילה את הקבצים הבאים בלבד: (כאשר ליד כל קובץ מצוין מספר המתאר את גודלו).

```
F1 6
d1/F1 8
d1/F2 10
d1/d2/F1/G 3
d2/F2/d3/A 2
d3/F2 4
```

לאחר הפעלת התוכנית ע"י הפקודה:

```
>P5 d F1 F2 A
```

יתקבל הפלט:

```
F1 6
F2 4
A 2
```

פתרון שאלה 5

```
#!/bin/bash
dir="$1"
shift
for fname in "$@"
do
  size_list=""
  find "$dir" -name "$fname" -type f >| tmp
  for file in $(cat tmp)
  do
    size=$(cat $file | wc -c)
    size_list="$size_list $size"
  done
  min_size=$(echo $size_list | tr " " "\n" | sort -n | head -1)
  echo "$fname $min_size"
done
```

# בהצלחה !