

דוגמאות שהוצגו בהרצאה 9 בקורס יסודות מערכות פתוחות

דוגמה 1

דוגמאות של שימוש בפקודה `getline` על כל הסוגים שלה שהראתי בכיתה נמצאות בסרטון 9 חלק 1. דוגמאות דומות לאלו שהראיתי בכיתה נמצאות בחוברת בעמודים 187-195.

דוגמה 2

דוגמאות של כתיבה לקובץ על ידי הפקודה `print` נמצאות בעמודים 195-197 בחוברת.

דוגמה 3

התכנית P1 הבאה היא תכנית ב-`awk` שמקבלת קלט שם קובץ שמכיל מספרים (לא בהכרח מטריצה ריבועית) ומדפיסה לפלט את סכום המספרים בקובץ שגדולים מהמספר המתאים להם בשורה הראשונה. (אם למספר מסוים אין מספר מתאים לו בשורה הראשונה אז לא לוקחים אותו).

לדוגמה, עבור קובץ F1 שתכנו הוא:

```
10 8 16 4
2 8 17 3 60
12 9 4 5 2 7 8
```

לאחר הפעלת התכנית על ידי הפקודה:

```
P1 F1
```

מתקבל הפלט:

```
F1:43
```

הפלט הנ"ל התקבל מסכום המספרים: $17+12+9+5$

להלן התכנית:

```
#!/bin/awk -f
{   for (j=1;j<=NF;j++) {
        A[NR,j]=$j
    }
    B[NR]=NF
}
END {
    s=0
    for (i=1;i<=NR;i++){
        for (j=1;j<=B[i];j++){
            if (A[1,j]!=" " && A[i,j]>A[1,j]){
                s+=A[i,j]
            }
        }
    }
    print FILENAME": "s1
}
```

:טקסט טמררררר

```
#!/bin/awk -f
{   for (j=1;j<=NF;j++) {
        A[NR,j]=$j
    }
    B[NR]=NF
}
END {
    s=0
    for (i=1;i<=NR;i++){
        for (j=1;j<=B[i];j++){
            if (A[1,j]!=" " && A[i,j]>A[1,j]){
                s+=A[i,j]
            }
        }
    }
    print FILENAME": "s1
}
```

דוגמה 4

התכנית הבאה מרחיבה את התכנית הקודמת למספר קבצים.
לכל קובץ מודפסת שורה וסדר הקבצים בפלט הוא לפי סדר
הופעתם ברשימת הפרמטרים לתכנית.

לדוגמה,

נניח שהקובץ F1 מכיל:

```
10 8 16 4
2 8 17 3 60
12 9 4 5 2 7 8
```

נניח שהקובץ A1 מכיל:

```
10 20 30 40 50
11 12 13 44 15
8 25
```

נניח שהקובץ B1 מכיל:

```
1 2 3 4 5
5 4 3 2 1 5
3 3 3 3 3
```

לאחר הפעלת התכנית על ידי הפקודה:

```
P1 F1 A1 B1
```

מתקבל הפלט:

```
F1:43
A1:80
B1:15
```

להלן התכנית:

```
#!/bin/awk -f
{
    for (j=1;j<=NF;j++) {
        A[FILENAME,FNR,j]=$j
    }
    B[FILENAME,FNR]=NF
    C[FILENAME]=FNR
}
END {
for (x=1;x<ARGC;x++) {
    file=ARGV[x]
    s=0
    for (i=1;i<=C[file];i++){
        for (j=1;j<=B[file,i];j++){
            if (A[file,1,j]!=" " &&
                A[file,i,j]>A[file,1,j]){
                s+=A[file,i,j]
            }
        }
    }
    print file": "s
}
}
```

:טקסט טרמטר

```
#!/bin/awk -f
{
    for (j=1;j<=NF;j++) {
        A[FILENAME,FNR,j]=$j
    }
    B[FILENAME,FNR]=NF
    C[FILENAME]=FNR
}
END {
for (x=1;x<ARGC;x++) {
    file=ARGV[x]
    s=0
    for (i=1;i<=C[file];i++){
        for (j=1;j<=B[file,i];j++){
            if (A[file,1,j]!=" " &&
                A[file,i,j]>A[file,1,j]){
                s+=A[file,i,j]
            }
        }
    }
    print file": "s
}
}
```

דוגמה 5

התכנית הבאה היא כמו התכנית הקודמת בתוספת הדרישה שסדר הקבצים יהיה לפי מספרי הסכומים שלהם בסדר מספרי עולה.

לאחר הפעלת התכנית על ידי הפקודה:

```
P1 F1 A1 B1
```

מתקבל הפלט:

```
B1:15
```

```
F1:43
```

```
A1:80
```

להלן התכנית:

```
#!/bin/awk -f
{
    for (j=1;j<=NF;j++) {
        A[FILENAME,FNR,j]=$j
    }
    B[FILENAME,FNR]=NF
    C[FILENAME]=FNR
}
END {
for (x=1;x<ARGC;x++) {
file=ARGV[x]
s=0
for (i=1;i<=C[file];i++){
    for (j=1;j<=B[file,i];j++){
        if (A[file,1,j]!=" " &&
            A[file,i,j]>A[file,1,j]){
            s+=A[file,i,j]
        }
    }
}
print s" "file > "tmp"
}

system ("sort -n tmp >| tmp1")
while (getline<"tmp1"==1) {
    print $2":"$1
}
}
```

```
#!/bin/awk -f
{   for (j=1;j<=NF;j++) {
        A[FILENAME,FNR,j]=$j
    }
    B[FILENAME,FNR]=NF
    C[FILENAME]=FNR
}
END {
for (x=1;x<ARGC;x++) {
    file=ARGV[x]
    s=0
    for (i=1;i<=C[file];i++){
        for (j=1;j<=B[file,i];j++){
            if (A[file,1,j]!="" &&
                A[file,i,j]>A[file,1,j]){
                s+=A[file,i,j]
            }
        }
    }
    print s" "file > "tmp"
}
    system ("sort -n tmp >| tmp1")
    while (getline<"tmp1"==1) {
        print $2:"$1
    }
}
```

דוגמה 6

התכנית הבאה מבצעת אותה משימה כמו התכנית של דוגמה 5.
לאחר הפעלת התכנית על ידי הפקודה:

```
P1 F1 A1 B1
```

מתקבל הפלט:

```
B1:15
```

```
F1:43
```

```
A1:80
```

היתרון של תכנית זו הוא שהכל נעשה בשלב ה-BEGIN ומקבלים תכנית רגילה שקוראת מקבצים, בדומה ל-JAVA. להלן התכנית:

```
#!/bin/awk -f
BEGIN {
for (x=1;x<ARGC;x++){
  file=ARGV[x]
  i=1
  split("",A)
  while (getline <file==1){
    for (j=1;j<=NF;j++){
      A[i,j]=$j
    }
    B[i]=NF
    i++
  }
  n=i--
  s=0
  for (i=1;i<=n;i++){
    for (j=1;j<=B[i];j++){
      if (A[1,j]!="" && A[i,j]>A[1,j]) s+=A[i,j]
    }
  }
  print s" "file ">"tmp"
}
system ("sort -n tmp >| tmp1")
while (getline <"tmp1"==1){
  print $2":"$1
}
}
```

```

#!/bin/awk -f
BEGIN {
for (x=1;x<ARGC;x++){
  file=ARGV[x]
  i=1
  split("",A)
  while (getline <file==1){
    for (j=1;j<=NF;j++){
      A[i,j]=$j
    }
    B[i]=NF
    i++
  }
  n=i--
  s=0
  for (i=1;i<=n;i++){
    for (j=1;j<=B[i];j++){
      if (A[1,j]!="" && A[i,j]>A[1,j]) s+=A[i,j]
    }
  }
  print s" "file ">"tmp"
}
system ("sort -n tmp >| tmp1")
while (getline <"tmp1"==1){
  print $2":"$1
}
}

```