

דוגמאות שהוצגו בהרצאה 3 בקורס יסודות מערכות פתוחות

דוגמה 1

התכנית הבאה מדפיסה על המסך את התוכן של הקובץ F1

```
while read x; do
    echo "$x"
done<F1
```

התכנית בפורמט טקסט:

```
while read x; do
    echo "$x"
done<F1
```

דוגמה 2

התכנית הבאה דורסת את התוכן של הקובץ F1 ומכניסה לתוכו את הקלט מהמשתמש (עד שהמשתמש מקליד ctrl+d).

```
while read x;do
    echo "$x"
done>|F1
```

התכנית בפורמט טקסט:

```
while read x;do
    echo "$x"
done>|F1
```

דוגמה 3

התכנית הבאה דורסת את הקובץ F2 ומכניסה אליו את התוכן של הקובץ F1.

```
while read x;do
    echo "$x"
done<F1 >|F2
```

התכנית בפורמט טקסט:

```
while read x;do
    echo "$x"
done<F1 >|F2
```

דוגמה 4

התכנית הבאה דורסת את הקובץ F2 ומכניסה אליו את התוכן של הקובץ F1 כאשר השורות של F1 משוכפלות (כל שורה מופיעה פעמיים).
התכנית בפורמט טקסט:

```
while read x;do
    echo "x"; echo "$x"
done<F1 >|F2
```

```
while read x;do
    echo "x"; echo "$x"
done<F1 >|F2
```

דוגמה 5

התכנית הבאה מדפיסה את המספרים 1 עד 8 ללא המספר 5.
מדגימה שימוש ב-`.continue`.

```
for x in $(seq 8);do
    if [ $x -eq 5 ]; then
        continue
    fi
    echo $x
done
```

התכנית בפורמט טקסט:

```
for x in $(seq 8);do
    if [ $x -eq 5 ]; then
        continue
    fi
    echo $x
done
```

דוגמה 6

התכנית הבאה מדפיסה את המספרים 1 עד 4.
מדגימה שימוש ב-`break`.

```
for x in $(seq 8);do
    if [ $x -eq 5 ]; then
        break
    fi
    echo $x
done
```

התכנית בפורמט טקסט:

```
for x in $(seq 8);do
    if [ $x -eq 5 ]; then
        break
    fi
    echo $x
done
```

דוגמה 7

התכנית הבאה מבקשת מספר `i` מהמשתמש ומדפיסה את השורה
ה-`i` בקובץ `F1`.
ללא שימוש ב-`pipeline` הפתרון הוא:

```
echo -n "Enter number: ";read i
head -$i >| tmp
tail -1 tmp
```

בפורמט טקסט:

```
echo -n "Enter number: ";read i
head -$i F1 >| tmp
tail -1 tmp
```

עם שימוש ב-`pipeline` הפתרון הוא:

```
echo -n "Enter number: ";read i
head -$i F1 | tail -1
```

בפורמט טקסט:

```
echo -n "Enter number: ";read i
head -$i F1 | tail -1
```

דוגמה 8

התכנית הבאה מדפיסה את המספר הגדול ביותר בקובץ F1:

```
echo $(cat F1) | tr " " "\n" | sort -n | tail -1
```

בפורמט טקסט:

```
echo $(cat F1) | tr " " "\n" | sort -n | tail -1
```

דוגמה 9

התכנית הבאה יוצרת קובץ F2 שמכיל את הקובץ F1 לאחר צמצום רווחים. (אם F2 היה קיים אז הוא נדרס).

```
while read x;do
    echo x;
done<F1 >|F2
```

בפורמט טקסט:

```
while read x;do
    echo x;
done<F1 >|F2
```

דוגמה 10

כתוב תכנית שמוצאת את גודל הקובץ הגדול ביותר בתיקיה הנוכחית מבין כל הקבצים ששמן מתחיל באות F. התכנית מוסברת בעמודים 93-95 בחוברת.

```
x=$(echo $(wc -l F*) | wc -w)-2]
y=$(seq 1 2 $x)
z=$(echo $y | tr " " ",")
echo $(wc -l F*) | cut -d" " -f$z | \
tr " " "\n" | sort -n | tail -1
```

בפורמט טקסט:

```
x=$(echo $(wc -l F*) | wc -w)-2]
y=$(seq 1 2 $x)
z=$(echo $y | tr " " ",")
echo $(wc -l F*) | cut -d" " -f$z | \
tr " " "\n" | sort -n | tail -1
```