

דוגמאות שהוצגו בהרצאה 11 בקורס יסודות מערכות פתוחות

דוגמה 1

הדוגמה הבאה מדגימה מבנה של פקודת `expect` שמצפה לכמה אפשרויות ומגיבה באופן שונה לכל אחת מהאפשרויות.

להלן תכנית בשם `P2` שקולטת מהמשתמש מחרוזת. אם המשתמש מקליד `sum` התכנית קולטת 2 מספרים ומדפיסה את סכומם. אם המשתמש מקליד `concatenate` התכנית קולטת 2 מחרוזות ומדפיסה את השרשור שלהן. אם המשתמש מקליד מחרוזת אחרת, התכנית מדפיסה הודעת שגיאה.

להלן התכנית `P2`:

```
echo -n "What do you want to do? ";read x
case $x in
"sum")
    echo -n "enter num 1: "; read n1
    echo -n "enter num 2: "; read n2
    echo The sum of $n1 and $n2 is:[$n1+n2];;
"concatenate")
    echo -n "enter string 1: "; read s1
    echo -n "enter string 2: "; read s2
    echo The concatenation of $s1 and $s2 is:$s1$s2;;
*)
    echo You had to type sum or concatenate;;
esac
```

ובפורמט טקסט:

```
echo -n "What do you want to do? ";read x
case $x in
"sum")
    echo -n "enter num 1: "; read n1
    echo -n "enter num 2: "; read n2
    echo The sum of $n1 and $n2 is:[$n1+n2];;
"concatenate")
    echo -n "enter string 1: "; read s1
    echo -n "enter string 2: "; read s2
    echo The concatenation of $s1 and $s2 is:$s1$s2;;
*)
    echo You had to type sum or concatenate;;
esac
```

התכנית P1 הבאה היא תכנית expect שמפעילה את התכנית P2 הנ"ל ופועלת באופן שונה בהתאם למחרוזת שהתכנית P1 קיבלה בפרמטר הראשון.

לאחר הפעלת התכנית P1 על ידי הפקודה:

```
P1 sum 10 20
```

מתקבל הפלט:

```
spawn P2
```

```
What do you want to do? sum
```

```
enter num 1:10
```

```
enter num 2:20
```

```
The sum of 10 and 20 is:30
```

לאחר הפעלת התכנית P1 על ידי הפקודה:

```
P1 concatenate 10 20
```

מתקבל הפלט:

```
spawn P2
```

```
What do you want to do? concatenate
```

```
enter string 1:10
```

```
enter string 2:20
```

```
The concatenation of 10 and 20 is:1020
```

לאחר הפעלת התכנית P1 על ידי הפקודה:

```
P1 zzz 10 20
```

מתקבל הפלט:

```
spawn P2
```

```
What do you want to do? zzz
```

```
You had to type sum or concatenate
```

להלן התכנית P1:

```
#!/usr/bin/expect
set what_to_do [lindex $argv 0]
set arg1 [lindex $argv 1]
set arg2 [lindex $argv 2]
spawn P2
expect What {send "$what_to_do\n"}
expect {
  "num 1" {send "$arg1\n";
           expect "num 2" {send "$arg2\n"}}
  "string 1" {send "$arg1\n";
              expect "string 2" {send "$arg2\n"}}
  "You" {exit}
}
expect "The" {exit}
```

:טקסט טברמט 19107

```
#!/usr/bin/expect
set what_to_do [lindex $argv 0]
set arg1 [lindex $argv 1]
set arg2 [lindex $argv 2]
spawn P2
expect What {send "$what_to_do\n"}
expect {
  "num 1" {send "$arg1\n";
           expect "num 2" {send "$arg2\n"}}
  "string 1" {send "$arg1\n";
              expect "string 2" {send "$arg2\n"}}
  "You" {exit}
}
expect "The" {exit}
```

דוגמה 2

נניח שיש תכנית P5 שתכנה הוא:

```
echo $x  
x=10
```

ונניח שהתכנית P1 הבאה קוראת לתכנית P5

```
x=8  
P5  
echo $x
```

מאחר והקריאה ל-P5 יוצרת תת תהליך עם משתנים שלו
ההצבה של $x=10$ בתכנית P5 לא משפיעה על המשתנה x
בתכנית P1.

לאחר הפעלת התכנית P1 מתקבל הפלט:

שורה ריקה

8

לעומת זאת אם התכנית P1 תקרא ל-P5 עם נקודה (ורווח)
לפני הקריאה, המשמעות היא שהקוד של P5 נשתל במקום
הקריאה. לכן אם התכנית P1 תראה כך:

```
x=8  
. P5  
echo $x
```

אזי, לאחר הפעלת התכנית P1 יתקבל הפלט:

8

10

דוגמה 3

מיון לפי שדות על ידי הפקודה `.sort -k`
דוגמאות עמודים 230-233.

דוגמה 4

הפקודה `.rev` דוגמאות בעמוד 222.

למשל לאחר הפעלת:

```
echo abcd | rev
```

מתקבל הפלט:

```
dcba
```

דוגמה 5

הרחבת סוגריים מסולסלים. דוגמאות בעמודים 223-224.
לאחר הפעלת קטע הקוד הבא:

```
echo {a,b}{c,d,e}
echo {a,b}{c,d,e}g
echo {a,b}{c,d}{e,f}
echo {a..d}
echo {a..d}{0-3}
```

ובפורמט טקסט:

```
echo {a,b}{c,d,e}
echo {a,b}{c,d,e}g
echo {a,b}{c,d}{e,f}
echo {a..d}
echo {a..d}{0..3}
```

מתקבל הפלט:

```
ac ad ae bc bd be
acg adg aeg bcg bdg beg
ace acf ade adf bce bcf bde bdf
a b c d
a0 a1 a2 a3 b0 b1 b2 b3 c0 c1 c2 c3 d0 d1 d2 d3
```

דוגמה 6

הפקודה `eval` . דוגמאות בעמודים 224-225.

לאחר הפעלת קטע הקוד הבא:

```
x=3
y=8
echo {$x..$y}
eval echo {$x..$y}
```

ובפורמט טקסט:

```
x=3
y=8
echo {$x..$y}
eval echo {$x..$y}
```

מתקבל הפלט:

```
{3..8}
3 4 5 6 7 8
```

דוגמה 7

פעולות על מחרוזות. דוגמאות בעמודים 217-221

לאחר הפעלת קטע הקוד הבא:

```
x=abcdef
echo ${#x}
echo ${x:2:3}
echo ${x:2}
for i in $(seq 0 1 ${#x}-1);do
  echo ${x:$i:1}
done
```

ובפורמט טקסט:

```
x=abcdef
echo ${#x}
echo ${x:2:3}
echo ${x:2}
for i in $(seq 0 1 ${#x}-1);do
  echo ${x:$i:1}
done
```

מתקבל הפלט:

```
6
cde
cdef
a
b
c
d
e
f
```

לאחר הפעלת קטע הקוד הבא:

```
x=abcdefg
echo ${x#a??*}
echo ${x##a??*}
echo ${x#b??*}
echo ${x##b??*}
echo ${x%*fg}
echo ${x%*f}
echo ${x%%*f}
echo ${x%%*fg}
```

ובפורמט טקסט:

```
x=abcdefg
echo ${x#a??*}
echo ${x##a??*}
echo ${x#b??*}
echo ${x##b??*}
echo ${x%*fg}
echo ${x%*f}
echo ${x%%*f}
echo ${x%%*fg}
```

מתקבל הפלט:

```
defg
שורה ריקה
abcdefg
abcdefg
abcde
abcdefg
abcdefg
שורה ריקה
```

לאחר הפעלת קטע הקוד הבא:

```
x=cababcdab
echo ${x/ab/zzz}
echo ${x/ab*/zzz}
echo ${x//ab/zzz}
echo ${x//ab*/zzz}
```

ובפורמט טקסט:

```
x=cababcdab
echo ${x/ab/zzz}
echo ${x/ab*/zzz}
echo ${x//ab/zzz}
echo ${x//ab*/zzz}
```

מתקבל הפלט:

```
czzzabcdab
czzz
czzzzzzcdzzz
czzz
```

דוגמה 8

הפקודה `.shift` דוגמאות בעמוד 129.

דוגמה 9

מערכים ב-`bash` דוגמאות עמודים 230-226.

לאחר הפעלת קטע הקוד הבא:

```
a=(aa bb cc dd ee)
echo ${a[0]}
a[0]=zz
echo ${a[0]}
echo ${a[@]}
a[0]="zz ww"
echo ${a[@]}
echo "${a[@]}"
echo $#a[@]
echo "${#a[@]}"
```


ובפורמט טקסט:

```
a=(aa bb cc dd ee)
echo ${a[0]}
a[0]=zz
echo ${a[0]}
echo ${a[@]}
a[0]="zz ww"
echo ${a[@]}
echo "${a[@]}"
echo ${#a[@]}
echo "${#a[@]}"
```

מתקבל הפלט:

```
aa
zz
zz bb cc dd ee
zz ww bb cc dd ee
zz ww bb cc dd ee
5
5
```

לאחר הפעלת קטע הקוד הבא:

```
a=(aa bb cc)
echo ${a[@]}
a=8
echo ${a[@]}
unset a
echo ${a[@]}
a=8
a[1]=7
a[6]=zz
echo "${a[@]}"
```

ובפורמט טקסט:

```
a=(aa bb cc)
echo ${a[@]}
a=8
echo ${a[@]}
unset a
echo ${a[@]}
a=8
a[1]=7
a[6]=zz
echo "${a[@]}"
```

מתקבל הפלט:

```
aa bb cc
8 bb cc
שורה ריקה
8 7 zz
```

דוגמה 10

שאלת הספרייה שהופיעה במועד א 2016 והפתרון שלה מוצג
בפתרון שאלה 11.1 בתרגילי הבית של 2017.