

בקרת תהליכים

❖ כדי להציג את הסטטוס של התהליכים הרצים
אפשר להשתמש בפקודה: **ps [options]**

❖ ps:

```
shani@shani-VirtualBox:~/Desktop$ ps
  PID TTY          TIME CMD
 1995 pts/0        00:00:00 bash
 4600 pts/0        00:00:00 ps
```

The difference between a "job" and a "process"

- ❖ A **process** is any running program with its own address space.
- ❖ A **job** is a concept used by the shell - any program you interactively start that doesn't detach (i.e., under the direct control of a user) is a job.
 - The shell has an identifier number to each job (which is **different** from the PID)

בקרת תהליכים

❖ מספר תהליכים רצים על המערכת, בטרמינל השולט רק תהליך אחד מתקשר עם המקלדת והמסך – foreground job.

❖ שאר התהליכים נקראים background jobs

❖ כדי להריץ תהליך ברקע (למה?) נשתמש ב "&" לאחר הרצת התהליך:

❖ ./a.out &

בקרת תהליכים

❖ שליחת תהליך לרקע:

- שלב א': עצירת התהליך CTRL+Z
- שלב ב': המשך התהליך ברקע בעזרת הפקודה: bg

```
# ./a.out
```

```
# [CTRL-Z]
```

```
# bg
```

בקרת תהליכים

❖ הצגת כל התהליכים שרצים ברקע תעשה על ידי שימוש בפקודה jobs

jobs

[1] Running bash download-file.sh &

[2]- Running evolution &

[3]+ Done nautilus .

תהליך אחד לפני אחרון
שהועבר לרקע

התהליך האחרון
שהועבר לרקע.

בקרת תהליכים

jobs [-Inprs] [JOBSPEC ...]

JOBSPEC

Job name or number.

- l** lists process IDs in addition to the normal information.
- n** list only processes that have changed status since the last notification.
- p** lists process IDs only.
- r** restrict output to running jobs.
- s** restrict output to stopped jobs.

דוגמא

❖ כאשר יש לי 9 תהליכים בשם ./a.out. המורצים ברקע.

❖ לאחר ששבעת הראשונים סיימו לרוץ, אם נשתמש בפקודה jobs נראה :

```
shani@shani-VirtualBox:~/Dropbox/Operating Systems/Tirgul/T5$ jobs  
[8]-  Running                  ./a.out &  
[9]+  Running                  ./a.out &
```

❖ אם נבקש לקבל את התהליך לפי שמו נתקל בבעיה :

```
shani@shani-VirtualBox:~/Dropbox/Operating Systems/Tirgul/T5$ jobs 8  
[8]-  Running                  ./a.out &
```

בקרת תהליכים

❖ הבאת תהליך מהרקע:

- שלב א': הצגת כל התהליכים הרצים ברקע
- שלב ב': בחירת התהליך בעזרת הפקודה: `fg %[num]`

```
# jobs
```

```
[1] Running bash download-file.sh &
```

```
[2]- Running evolution &
```

```
[3]+ Done nautilus .
```

```
# fg %1
```

התהליך האחרון שהועבר
לרקע (מסומן על ידי "+"),
יהיה הראשון שיחזור
במידה ונשתמש בפקודה
`fg` בלי ארגומנטים