

PART 2B

A. כל מספר $1 \leq x \leq 101$ יודפס 2^x פעמים כך: $x = 2^x$, אך הסדר אינו ידוע. לא ניתן לדעת מתי כל פרוסס ירוץ ומי ידפיס קודם. ניתן להבטיח ש $x=1$ כלשהו (מתוך ה-2) יודפס ראשון כי הפרוסס המקורי או הבן של המקורי יריצו את הפקודה `print` לפני `++x`. כמו כן, מובטח ש $x=101$ כלשהו (מתוך 2^{101} פעמים) יודפס אחרון כי איזשהו פרוסס תמיד יהיה אחרון ותייה לו את הפקודה `print` האחרונה עם $x=101$.

B. 2 השורות הבאות יודפסו ראשונות (לא ידוע מי הראשונה והשנייה), PID1 הוא הבן, PID2 הוא האבא:

PID PID1: Pending for signal.

PID PID2: Pending for signal.

לאחר מכן יודפס:

PID1: Caught User signal from another process

PID2: Caught User signal from another process

PID1: Caught Interrupt, exiting now.

ה KILL הרביעי `kill -s USR1 PID1` לא ישפיע על ההדפסות היות ו PID1 כבר מת ולא רץ.

לאחר מכן יודפס:

PID2: Caught Interrupt, exiting now.

PART 3

1. לא נכון. לאחר ביצוע פקודת `fork`, כל אחד מהפרוססים (האבא והבן) הוא פרוסס בפני עצמו. הריגת האבא לא משפיעה על הבן ולכן הבן ימשיך לרוץ.
2. לא נכון. כל תהליך יכול להשתמש ב `named pipe`. גם פרוססים `non-related` וגם `related`.
3. לא נכון. קריאה ל `start` תיצור `thread` חדש בהכרח ולאחר מכן תריץ את `run` תחת ה `thread` החדש. לעומת הרצת `run` רק מריצה את `run` תחת אותו `thread` שקרא לו ללא יצירת `thread` חדש.