

### שאלה 1:

נוצרים ארבעה תהליכים חדשים.

הסבר: תהליך האב מבצע את פקודת fork ומייצר בן ראשון.

פעולת fork מחזירה שני ערכים: עבור תהליך האב נקבל פסוק אמת מכיוון שהוא מקבל את ה-pid של הבן שיצר (ששווה מאפס) ועבור הבן נקבל פסוק שקר (כי הערך המוחזר מ-fork עבורו שווה לאפס). מכאן שערכו של a אצל האב שווה ל-2 וערכו של a אצל הבן הראשון שווה ל-3. בפקודה הבאה נקבל פסוק אמת רק עבור האב ולכן הוא מייצר בן שני שערכי המשתנים שלו זהים לערכים של האב. מכאן שערכו של a אצל האב והבן השני שווה ל-2 וערכו של a אצל הבן הראשון שווה ל-3. בפקודה הבאה ערכי ה-a של שלושת התהליכים גדלים באחד. בשורה האחרונה נקבל פסוק אמת עבור האב ולכן הוא מייצר את הבן השלישי, וגם נקבל פסוק אמת עבור הבן השני שמייצר בעצמו את התהליך הרביעי.

### שאלה 2:

יכולות להתקבל תוצאות שונות בכל הרצה, כתלות בחלוקת זמן המעבד בין התהליכים. לדוגמא:

PARENT: x is 150  
PARENT: forking...  
PARENT: forked...  
PARENT: forked...  
CHILD: happy birthday  
CHILD: 300  
PARENT: child completed  
PARENT: 450

דוגמא נוספת לפלט אפשרי:

PARENT: x is 150  
PARENT: forking...  
PARENT: forked...  
CHILD: happy birthday  
CHILD: 300  
PARENT: forked...  
PARENT: child completed  
PARENT: 450

הסבר: בתחילת התכנית תהליך האב רץ, הוא מדפיס את ערכו של x ואת השורה forking. לאחר מכן הוא מבצע fork ונוצר לו תהליך בן שרץ על התכנית בדיוק מאותה הנקודה. מכאן האופציות להמשך הן רבות כתלות בחלוקת זמן המעבד בין התהליכים. במידה והבן מקבל מספיק זמן לבצע את כל פעולותיו לפני האב נקבל את הפלט של הדוגמא השנייה. במידה והזמן מחולק אחרת נוכל גם לקבל הדפסות מקוטעות ומשולבות בין שני התהליכים. חשוב להדגיש שהדפסה שהילד סיים נעשית רק כאשר הילד באמת סיים את הריצה שלו מכיוון שהאב מחכה שיסיים ורק אז מדפיס. בהקשר לערכי ההדפסות: מכיוון שהבן הוא העתק של האב, ערכו של המשתנה x אצל הבן הוא 150 והוא מכפיל אותו פי 2 ולכן מדפיס 300. כאשר תהליך הבן מסתיים תהליך האב ממשיך, הוא מדפיס את ההודעה שהילד סיים, ערכו של x אצל האב לא השתנה בעקבות השינוי שהבן ביצע על המשתנה x שלו מכיוון שלבן נוצר משתנה אחר ברגע שהוא מבצע שינוי, ולכן כאשר הוא מכפיל את x פי 3 אנחנו מקבלים 450.

**quitting**  
**2**

**הסבר:** תהליך האב מייצר תהליך בן, הערך child אצל האב שווה לpid של הבן ששונה מאפס, בעוד שערך הchild אצל הילד שווה לאפס. שניהם ממשיכים לרוץ במקביל לתוך הswitch, תהליך האב מגיע לפקודת default שאומרת לו לחכות עד שתהליך הבן יסתיים, במקביל הבן נכנס ל case 0 שם הוא מדפיס **quitting** ומבצע יציאה עם ערך החזרה. מכיוון שתהליך הבן סיים, תהליך האב ממשיך, הוא מדפיס את סטטוס היציאה של הבן שכאמור הוא **2** ומסיים גם הוא.

**4**

**הסבר:** הערך ההתחלתי של value הוא 3, תהליך האב מייצר תהליך בן, ושניהם נמצאים בתוך if, מכיוון שהערך המוחזר מפעולת fork אצל האב, הוא pid של הבן ששונה מאפס, מתקבל אמת והוא נכנס לתוך התנאי שבו הוא מחכה לתהליך הבן. במקביל מכיוון שהערך המוחזר מfork אצל הבן הוא שווה לאפס, הוא נכנס לelse, ומסיים את ריצתו כאשר הוא מחזיר את הערך 3. האב ממשיך והוא מכניס לתוך value את הערך שהבן החזיר כאשר הוא יצא כלומר את המספר 3. מכיוון שפקודה לאחר מכן נעשה ++value הוא גדל ל**4** ולכן זה מה שיודפס למסך.