

שאלה 1

בהנחה שה-*fork* צלח, תהליך האבא יחכה לתהליך הבן, יאסוף את ערך ההחזרה שלו (*wait(&value)*) ויוסיף אליו 1. תהליך הבן מסיים עם ערך ההחזרה *exit(value)*. לכן המשתנה *value* יעודכן לערך $4 = 3 + 1$. **ופלט התוכנית יהיה 4.**
מהדוקומנטציה של הפקודה *WEXITSTATUS(value)*, היא מחזירה את קוד היציאה שהוחזר על-ידי תהליך הבן ועל כן התשובה.

שאלה 2

הזיכרון הסטטי אינו משותף בין תהליכים, לכן אע"פ שהמשתנה גלובאלי, הוא אחר מזה של תהליך האב. כאשר ה-*pid* שונה מאפס זאת אומרת שאנו בתהליך האבא, כלומר לא בתהליך הבן, ולכן המשתנה אינו מושפע מהוספת 15.
פלט התוכנית הינו:

Parent: value = 5

שאלה 3

נוצרים 4 תהליכים.

נוצר תהליך אחד בתנאי הראשון, וכעת יש שני תהליכים, אחד עם $a = 3$ לתהליך הראש ותהליך בן עם $a = 2$. תהליך הבן נכנס גם לתנאי השני ויוצר תהליך נוסף. כעת יש שלושה תהליכים, שניים עם $a = 2$ ותהליך הראשי עם $a = 3$. בשורה $a++$ הערך של כל התהליכים עולה – שני תהליכי הבן עם ערך $a = 3$ ותהליך האב עם ערך $a = 4$. שני התהליכים שנוצרו נכנסים לתנאי השלישי ויוצרים שני תהליכים נוספים. בסה"כ נוצרים 4 תהליכים במהלך התוכנית.

שאלה 4

נוצרים 127 תהליכים, לא כולל תהליך האבא.

כאשר $i = 1$ מתבצע *fork* פעמיים, פעם ראשונה מהלולאה ופעם שניה בתוך התנאי. בשלב זה יש $2^2 = 4$ תהליכים. כאשר $i = 2$ אותו הדבר קורה והפעם נסיים את הלולאה עם $2^4 = 16$ תהליכים. כאשר $i = 3$ מתבצע *fork* נוסף וכעת יש $2^5 = 32$ תהליך. עבור $i = 4$ ישנם $2^6 = 64$ תהליכים ובאיטרציה האחרונה, כאשר $i = 5$, מספר התהליכים כולל התהליך הראשי הינו $2^7 = 128$. נחסיר את התהליך הראשי ולכן נוצרים 127 תהליכים.

במילים אחרות: ה-*fork* הראשון נקרא 5 פעמים והשני נקרא פעמיים, לכן סה"כ נקבל $2^7 - 1 = 127$ תהליכים.