

פתרון מבחן במערכות הפעלה סמסטר ב' מועד א' תשס"ג, 2 ביולי 2002 פתרון על ידי עמוס תנאי וסיון טולדו

הסבר לציון הסופי: לציון המבחן הוספה נקודה לכל תרגיל שהוגש וקיבל ציון סביר. למבחנים שנוקדו מתחת ל-60 אבל תוספת התרגילים העלתה את הציון הסופי ל-60 או יותר, עוגל ציון המבחן ברשומות הציונים ל-60 אבל תוספת הנקודות עבור התרגילים היא על בסיס ציון המבחן המדויק. תלמידים שגם לאחר תוספת הניקוד על התרגילים לא הגיעו לציון 60 נכשלו בקורס, והציון הסופי שלהם שווה לציון המבחן.

שאלה 1: 20 נקודות (5 לכל סעיף)

1. נכון, שני תהליכים עלולים להקלע לקיפאון אם אחד פותח את קובץ א' ומנסה לאחר מכן לפתוח קובץ ב', שפתוח כבר על ידי תהליך אחר, שמנסה לפתוח את א'. האיתותים לא רלוונטיים לשאלה (A): הפחתה של 2 נק' בגין תשובה שטוענת שהאיתותים עוזרים במשהו לבעיה).
2. ניתן להרוג בעזרת הפקודה כל תהליך של המשתמש. root יכול להרוג כל תהליך, אבל תהליכים של משתמשים רגילים לא יכולים להרוג תהליכים של משתמשים אחרים. A: הפחתה של 2 נק' על תשובה שלא מסייגת ביחס למשתמשים אחרים, B: הפחתה של 2 נק' לתשובה שמבלבלת בין מצב מיוחס ובין המשתמש שמריץ את התהליך, C: הפחתה של 4 נק' על תשובה שטוענת שניתן להתעלם מהאיתות.
3. שני האירועים האפשריים הם קבלת הודעת reset מהצד השני (הצד השני מדווח בכך שלדעתו הקשר כבר סגור) או פקיעת טיימר, שמלמדת בדרך כלל שהקשר הפיזי עם הצד השני נותק. אירוע נוסף סביר הוא שרמה נמוכה יותר במחסנית הפרוטוקולים מזהה בעיה, למשל ניתוק של כבל רשת, מדווחת ל-TCP, ו-TCP מדווח על השגיאה בצורה זו לתהליך. A: הפחתה של 2 נק' בגין תשובה שמזכירה התנתקות רשת אבל ללא פקיעת טיימר או תגובה של רמה נמוכה יותר. B: הפחתה של 2 נק' על תשובה שמזכירה נפילה של תהליך בצד השני אבל ללא תיאור השתקפות האירוע במחשב שמחזיר את הודעת השגיאה.
4. לא, השגיאה אינה אמורה לקרות ב-UDP. קיבלנו תשובות שטוענות ששגיאה עלולה לקרות כתוצאה מדיווח של רמה נמוכה יותר במחסנית הפרוטוקולים על שגיאה.

שאלה 2: 30 נקודות

- א. העיקרון העיקרי במימוש מערכת קבצים כזו הוא ויתור על חציצה, כלומר שגרות מערכת הקבצים צריכות לכתוב למדיה מהר ככל האפשר בלוקים שהשתנו, ולא לעכב את כתיבתם. מזה נובע גם שיש להשתמש בשיטת הכתיבה הזזה, מכיון ששאר השיטות מיועדות לאפשר חציצה, לעיתים תוך שמירה על קונסיסטנטיות מבנה מערכת הקבצים. מכיון שאין משתמשים בחציצה, אין טעם בניהול יומן או עדכונים רכים, וכמובן שרצוי מאוד שלא להשתמש בכתיבה עצלה.

- ב. כאשר יש קבצים פתוחים על גבי המדיה הנשלפת, מערכת ההפעלה תאסור על שלפיה. אם כל הקבצים במדיה סגורים (כלומר לא בשימוש כעת), מערכת ההפעלה תבצע את הפעולות המצויינות בסעיף ג' ואז תתיר לשלוף את המדיה.
- ג. לפני שמערכת ההפעלה מתירה לשלוף את המדיה, היא דואגת לסיום כל הכתיבות למדיה. גם אם אין שימוש מכוון בחציצה, יתכן שהכתיבה האחרונה לא הסתיימה, ויתכן שאולי יש תור של כתיבות שממתינות לביצוע על ידי הבקר.

שגיאות נפוצות והפחתת נקודות בגינן:

- A. לא הוזכר בא' שיש להמנע ככל האפשר מחציצה בתוך מערכת ההפעלה או לכתוב מהר ככל האפשר. 5 נק'.
- B. לא הוזכר בא' שיש להשתמש בסדר כתיבות זהיר (כתיבה זהירה). 5 נק'.
- C. תשובה שטוענת שיש להשתמש במערכת מתאוששת או מערכת עדכונים רכים בא'. המערכות הללו נועדו לאפשר חציצה ולכן אינן רלוונטיות למקרה זה. 7 נק'.
- D. ב' ו/או ג': חסר קישור בין איסור שליפה ובין קבצים פתוחים. 8 נק'.
- E. ב' ו/או ג': חסר הסבר שמערכת הקבצים כותבת בלוקים מלוכלכים לפני אישור שליפה. 8 נק'.
- F. ב': איסור שליפה אם יש בלוקים מלוכלכים אבל אין קבצים פתוחים. 5 נק'.

שאלה 3: 35 נקודות

חלוקת נקודות: 14 לסעיף א', 7 לכל אחד משאר הסעיפים. להלן פתרון סעיף א':

```
HANDLE queue_global_lock;
HANDLE queue_global_condvar;
int queue_global_in_use_count;
struct {
    // similar, replace mutex_t and cond_t by HANDLE
} queue[25];
...
void nb_wait(int request) {
    WaitForSingleObject( queue[request].lock , INFINITE );
    while ( queue[request].done == FALSE ) {
        ResetEvent( queue[request].condvar );
        ReleaseMutex( queue[request].lock );
        WaitForSingleObject( queue[request].condvar , INFINITE );
        WaitForSingleObject( queue[request].lock , INFINITE );
    }
    queue[request].in_use = FALSE;
    ReleaseMutex( queue[request].lock );

    WaitForSingleObject( queue_global_lock , INFINITE );
    queue_global_in_use_count--;
    SetEvent( queue_global_condvar );
    ReleaseMutex( queue_global_lock );
}
```

להלן שגיאות נפוצות והפחתה בגינן. לא הופחתו נקודות בגין שגיאות כתיב בשמות השגרות ובגין שגיאות לוגיות לא טריויאליות.

- A. המנעולים והאירועים אינם מוגדרים כ-HANDLE. 5 נק'.
- B. שימוש בשמות שגרות לא נכונות (למשל שימוש ב-lock וכדומה). 8 נק'.
- C. חסר זמן פקיעה INFINITE בשגרות המתנה. 2 נק'.
- D. שגיאות לוגיות חמורות כגון בעיות ב-ResetEvent, SetEvent, בעיות בנעילות. עד 10 נק'.

סעיף ב'. המצב קורה לאחר שחרור האיבר request מהתור על ידי איפוס השדה in_use, אבל לפני ההפחתה מהמשתנה queue_global_in_use_count.

סעיף ג'. החזקת האיבר request כנעול עד להפחתת 1 מהמונה הגלובלי, במקום שחרורו מייד לאחר סימונו כפנוי.

סעיף ד'. המימוש אינו לקוי, אך הוא עלול לגרום להמתנה מעט ארוכה מדי. למעשה ברגע שמסמנים את האיבר כפנוי ניתן להודיע על האירוע הגלובלי, משום שכעת מצב התור אולי השתנה, מכיון שהוא אינו מלא לחלוטין.

שאלה 4: 15 נקודות

החלטת ניתוב		קבוצת יעדים	
כתובת IP	כרטיס רשת	מסיכה (mask)	כתובת IP
10.0.0.2	eth1	0.0.0.0	0.0.0.0
ישירות	eth0	255.255.255.248	132.67.110.0
132.67.110.5	eth0	255.255.255.254	132.67.110.6

השורה הראשונה בטבלה היא ניתוב ברירת מחדל (default gateway), ומכיון שכל הסיביות במסיכה כבויות, היא מתאימה לכל כתובת. היא מורה לנתב דרך כרטיס הרשת eth1 ודרך הנתב הימני באיור.

השורה השנייה מורה לנתב לכל הכתובות 132.67.110.0 עד 132.67.110.7 דרך כרטיס הרשת eth0, ישירות (כלומר ללא מעבר במחשב או נתב ביניים). החלטת הניתוב הזו מציינת תחילית של 29 סיביות, היא יותר ספציפית מהשורה הקודמת, ולכן כתובת בטווח הזה לא ישתמשו בניתוב ברירת המחדל.

השורה השלישית היא החלטת ניתוב עוד יותר ספציפית, עם תחילית של 31 סיביות, שמתארת ניתוב לשתי כתובות בדיוק, 132.67.110.6 ו-132.67.110.7, והיא מורה לנתב אליהן דרך המחשב שכתובת ה-IP שלו היא 132.67.110.5.

על שגיאות שקשורות אך ורק לניתוב לכתובת 132.67.110.0 לא הופחתו נקודות, מכיון שזו כתובת מיוחדת מסוג שלא נדון בכיתה. כלומר תשובות שהניחו שהכתובת הזו שייכת אולי למחשב ב-"שאר העולם" התקבלו, וגם תשובות שמתעלמות מניתוב לכתובת הזו.

טעויות נפוצות והפחתת הנקודות בגינן:

- A. שורה עבור כל מחשב שמחובר למקטע האתרנט. 4 נק'.
- B. חסר ניתוב לכתובת 132.67.110.6. 2 נק'.
- C. שורה עבור המחשב הימני העליון 132.67.110.10. 3 נק'.
- D. הבנה לא נכונה של תפקוד המסיכה. 2 נק'.
- E. חסר ניתוב ברירת מחדל. 4 נק'.