

22.2.2007

תכנות מתקדם 61617
מבחן סיום - מועד א'
 מרצה: ד"ר איתן רון

הערות

- (1) מותר כל חומר עזר.
- (2) זמן הבחינה - 3 שעות ללא אפשרות הארכה.
- (3) קרא בעיון את השאלות. אורכה של שאלה אינה בהכרח מעיד על הקושי שבה.
- (4) שאלות 1, 2, 3 הם שאלות תכנות. התשובות לשאלות חייבות להיות בקוד C או אסמבלי.

שאלה מספר 1_ (33 נקודות)

בשאלה זו עליך לממש רוטינת שרות הממשת "סינכרון" של שני תהליכים. שני תהליכים מסתנכרנים ע"י קריאה לאותה רוטינה sync והיא דואגת שהראשון שקורא לה ימתין לשני. הסינכרון מתבטא בכך שעם הקריאה של התהליך השני ל-sync שני התהליכים ימצאו במהלך קוד של חזרה מ-sync. עם הקריאה השניה ל-sync עליך לדאוג לשחרור המשאבים המשותפים. ניקוד מלא (8 נקודות הבדל) ינתן רק לפתרון שאיננו משתמש ב-busy wait.

ההכרזה של sync תהיה:

```
int sync(int key);
```

כאשר key ישמש כשם משני של המשאבים המשותפים. sync יחזיר 1 במקרה של סינכרון מוצלח ו-1 לכל מקרה של כשלון.

לדוגמא, הפלט של התוכנית הבאה:

```
int main()
{
  int pid;

  printf("Before fork\n\n");
  pid = fork();
  if (pid == 0)
  {
    sleep(5);
    printf("Child: Sync\n");
    if (sync(123456789) == 1)
      printf("Child: After Sync\n");
    exit(0);
  } /* if */

  printf("Parent: Sync\n");
}
```

```

if (sync(123456789) == 1)
    printf("Parent: After Sync\n");

} /* main */

```

תהיה:

```

math:/home3/ronn > ./a.out
Before fork

Parent: Sync
Child: Sync
Child: After Sync
Parent: After Sync
math:/home3/ronn >

```

שאלה מספר 2 (34 נקודות)

א. (11 נקודות) כתוב תוכנית בשם decode.c המקבלת קלט מה-standard input, צוברת אותו לתוך מערך (ההנחה היא שאינה ארוכה יותר מ-4096 בתים) ועם סיומו מדפיסה ל-standard output את הפענוח שלו בהנחה שהתוכן הינו קובץ מוצפן ב"צופן הקיסרי" של 100 כלומר שכל תו c שהתקבל היה תוצאה של הסכום

$$c = (c'+100) \text{ modulo } 256$$

כלומר ש-decode.c מדפיס את ה-c' ולא את ה-c.

ב. (23 נקודות) כתוב תוכנית read_and_decode.c הקוראת קלט מה-standard input, ומדפיסה ל-standard output את הפענוח שלו שחייב להעשות בעזרת קובץ בינארי בשם decode שהוא תוצאת קומפילציה של התוכנית decode.c של סעיף א'. כל זה בכדי לאפשר שימוש בתוכנת פענוח אחרת בעתיד.

שאלה מספר 3 (33 נקודות)

כתוב תוכנית awk בשם word_report.awk המקבלת שם של קובץ מדפיסה את המילים המופיעות בו ומספרי השורה שבהם המילים הללו מופיעות. ניקוד מלא (5 נקודות הבדל) ינתן לתוכנית שהדפסת המילים תהיה ממוינת.

מבחינת התוכנית "מילה" היא כמובן שהאינטרפרטר של AWK ממילא מבחין בהם כלומר "while" ו-"while" הן שתי מילים שונות.

לדוגמא, אם אני אפעיל את התוכנית על הקובץ:

```

/* calc.c */

int main()
{
    int x;
    int y;

```

```

printf("Enter two integers:\n");
scanf("%d %d", &x, &y);

printf("%d + %d = %d\n", x, y, x+y);
printf("%d - %d = %d\n", x, y, x-y);
printf("%d * %d = %d\n", x, y, x*y);
printf("%d / %d = %d\n", x, y, x/y);

} /* main */

```

הפלט יהיה:

```

+: 11
=: 11 12 13 14
/*: 1 17
*/: 1 17
-: 12
*: 13
/: 14
}: 17
{: 4
calc.c: 1
%d: 11 12 13 14
%d",: 9
%d\n",: 11 12 13 14
int: 3 5 6
integers:\n");: 8
main: 17
main(): 3
printf("%d: 11 12 13 14
printf("Enter: 8
scanf("%d: 9
two: 8
x,: 11 12 13 14
x;: 5
&x,: 9
x+y);: 11
x-y);: 12
x*y);: 13
x/y);: 14
y,: 11 12 13 14
y;: 6
&y);: 9

```

שאלה מספר 4 (בונוס 5 נקודות)

כתוב סקריפט בשם `most_recent_file` המדפיס את השם של הקובץ הכי מאוחר בספרייה הנוכחית, כלומר הקובץ שעבר את העדכון האחרון מבין הקבצים בספרייה הנוכחית. הנח רשאי להניח שהספרייה אינה ריקה.

לדוגמא, הפלט של התסריט הבא:

```

% ls > ttt
% ./most_recent_file
Most recent file is ttt
%

```