

19.2.2004

מבחן מועד א'
מערכות פתוחות
סמסטר חורף, תשס"ד

- משך המבחן: שלוש וחצי שעות.
- יש לענות על כל השאלות.
- מותר השימוש בחומר עזר כלשהו, פרט למחשבים.
- יש להקפיד על כתיבה ברורה ומסודרת של התשובות.

1. (10 נקודות)

נתון קובץ script ב-Tcl בשם P1 שתוכנו:

```
#!/usr/bin/tclsh
# prog P1
proc f1 {C} {
  upvar 1 A a
  set a(1) {{1 2} {3 4}}
  f2 $C
}
proc f2 {A} {
  upvar 1 a d
  upvar 2 $A e
  puts "\$d(1)=$d(1)"
  puts "\$e(2)=$e(2)"
  set e(2) 8
}
set A(1) X
set X(1) 5
set X(2) 6
set Z [f1 $A(1)]
puts "Z=$Z"
puts "A(1)=$A(1) X(1)=$X(1) X(2)=$X(2)"
```

כתוב את הפלט שיתקבל (על המסך) לאחר הפעלת הקובץ משורת הפקודה
ע"י הפקודה:

> P1

2. (25 נקודות)

נגדיר "מערך משכורות של עובדים" כמערך אסוציאטיבי שבו האינדקסים הם שמות של עובדים (בפורמט: שם משפחה, שם פרטי) והערכים הם מספרים המציינים את משכורות העובדים.

לדוגמה קטע הקוד הבא ב-Tcl מגדיר "מערך משכורות של עובדים" בשם A שמכיל משכורות של ארבעה עובדים:

```
%set A(Reuven,Atar) 1000
%set A(Yosi,Ben-Ayun) 900
%set A(Ronen,Harzi) 1000
%set A(Eyal,Berkovitz) 1200
```

כתובי פונקציה ב-Tcl בשם Sort_Array שמקבלת כפרמטר שם של "מערך משכורות של עובדים" ומדפיסה לפלט הסטנדרטי (דהינו למסך) את רשימת השמות של העובדים שבמערך ממוינת לפי משכורתיהם בסדר יורד. על פורמט ההדפסה להיות כמו בדוגמה שלהלן.

הנחי שהמערך ששמו מועבר לפונקציה כפרמטר תקין (אין צורך לבדוק את תקינותו) ונמצא רמה אחת מעל הפונקציה (כפי שמתואר בדוגמה שלהלן).

דוגמה:

יהי P3 קובץ ה-Script הבא (בו הושמטה הגדרת הפונקציה Sort_array).

```
#!/usr/bin/tclsh
# prog P3
proc Sort_Array {...} {
...
...
}
set A(Reuven,Atar) 1000
set A(Yosi,Ben-Ayun) 900
set A(Ronen,Harzi) 1000
set A(Eyal,Berkovitz) 1200
Sort_array A
```

לאחר הפעלת הקובץ P3 משורת הפקודה ע"י הפקודה:

> P3

יודפס הפלט הבא למסך:

```
Eyal Berkovitz 1200
Reuven Atar 1000
Ronen Harazi 1000
Yossi Ben-Ayun 900
```

3. (10 נקודות)

להלן קטע קוד שהינו התחלת של קובץ Script ב-Tk בשם Screen1.

```
#!/usr/bin/wish
# Screen1
wm title . "Main Window"
wm minsize . 250 250
frame .fr1 -width 1.2c -height 1.2c -bg yellow
frame .fr2 -width 1.2c -height 1.2c -bg red
frame .fr3 -width 1.2c -height 1.2c -bg green
frame .fr4 -width 1.2c -height 1.2c -bg blue
...
```

השלם/י את החלק החסר בקטע הקוד הנ"ל ע"י הוספת פקודות pack בלבד על מנת שלאחר הפעלת הקובץ Screen1 משורת הפקודה ע"י הפקודה:

>Screen1

יתקבל המסך הבא:



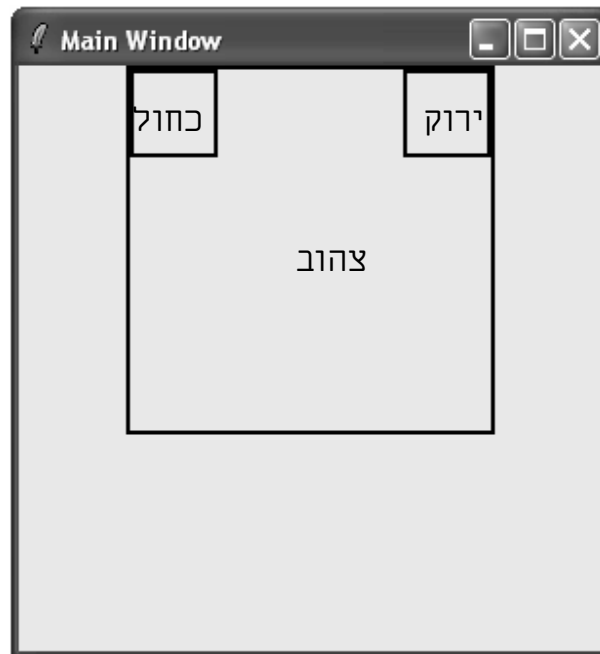
במסך הנ"ל 4 ריבועים בצבעים שונים. מאחר והבחינה אינה צבעונית צבעו של כל ריבוע צוין בתוך הריבוע (המסגרת השחורה מסביב לכל ריבוע היא להמחשה ואין צורך שהיא תופיע על המסך). לדוגמה הריבוע העליון בצבע צהוב והתחתון בצבע כחול.

4. (15 נקודות) נקודות

כתוב/י תוכנית (דהינו קובץ Script) ב-Tk בשם Screen2 שלאחר הפעלתה משורת הפקודה ע"י הפקודה:

>Screen2

יתקבל המסך הבא:

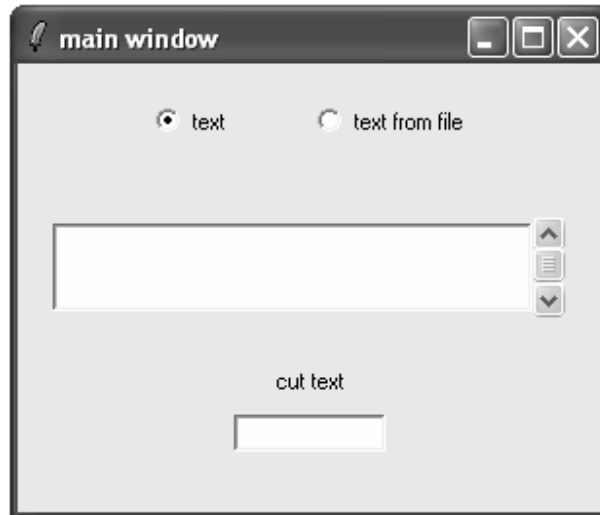


במסך הנ"ל 3 ריבועים בצבעים שונים. מאחר והבחינה אינה צבעונית צבעו של כל ריבוע צוין בתוך הריבוע (המסגרת השחורה מסביב לכל ריבוע היא להמחשה ואין צורך שהיא תופיע על המסך). הריבוע הגדול הינו בצבע צהוב וגודלו 5 x 5 ס"מ. הריבוע הצהוב מכיל שני ריבועים אחד כחול והשני ירוק שגודל כל אחד מהם הינו 1.2 x 1.2 ס"מ.

רמז: השתמש/י ביותר מ-3 מסגרות (דהינו frames) ליצירת המסך הנ"ל.

5. (40 נקודות)

כתוב/י תוכנית ב-Tk (דהינו קובץ script) בשם show_text שמקימת את הדרישות המתוארות בהמשך. על המסכים שהתוכנית תיצור להיות דומים ככל האפשר לאלה המוצגים בהמשך. עם הפעלת התוכנית מופיע המסך הבא:



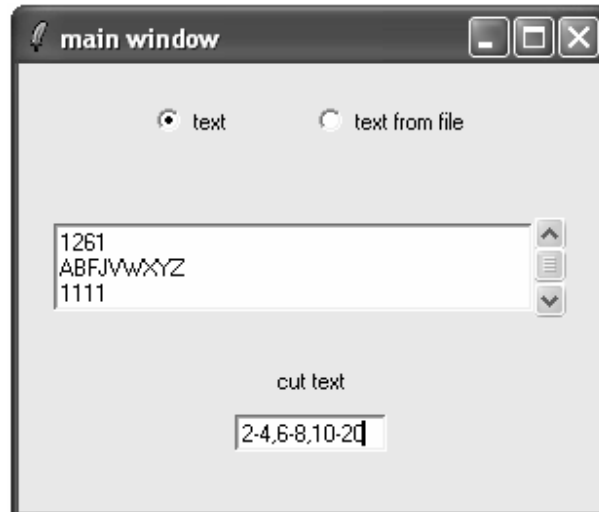
במסך זה נבחר הכפתור (מסוג radiobutton) שנקרא: "text". משמעות בחירה זו היא שתוכן חלון ה-text יהיה ריק כפי שמוצג במסך הנ"ל. המשתמש רשאי למלא את תוכן חלון ה-text כרצונו (אין מגבלה על מס' השורות). לדוגמה, לאחר שהמשתמש הקליד תווים לחלון ה-text הוא נראה כך:



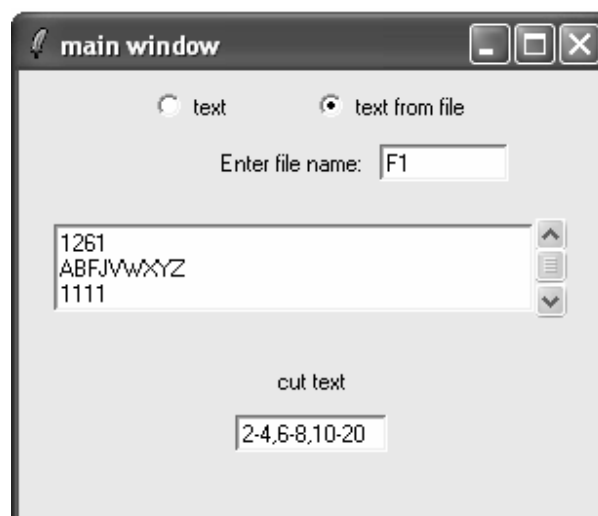
חלון ה-entry התחתון ימולא ע"י המשתמש באוסף של זוגות של מספרים (אחד או יותר ללא הגבלה על מס' הזוגות) בפורמט: מס'-מס'. זוגות המספרים מופרדים ע"י פסיקים. לדוגמה במסך הנ"ל המשתמש הקליד: 2-4,6-8,10-20

עם הקלדת מקש Return בחלון ה- entry כל שורה שמופיעה בחלון ה- text תחתך לפי זוגות המספרים שצוינו בחלון ה- entry. בדוגמה הנ"ל יושמטו מכל שורה תווים 2-4 (כולל 4) 6-8 ו- 10-20 (מספור התווים בכל שורה מתחיל מ-0). הנח/הניחי שכאשר המשתמש מקליד Return בחלון ה- entry אוסף הזוגות שבחלון הוא בפורמט חוקי ומספרי האינדקסים מסודרים בסדר עולה (אין צורך לבדוק את תקינות אוסף הזוגות).

לדוגמה לאחר הקלדת Return בחלון ה- entry שבמסך הקודם יופיע המסך הבא:



בחירת הכפתור (מסוג radiobutton) שנקרא: "text from file" תגרום להופעת חלון entry נוסף בו המשתמש יקליד שם קובץ (הנח/הניחי שהמשתמש יקליד שם קובץ שקיים במדריך בו נמצאת ומופעלת התוכנית show_text). לדוגמה, לאחר בחירת הכפתור "text from file" במסך הקודם והקלדת קובץ F1 בחלון ה- entry הנוסף שהופיע יתקבל המסך הבא:



לאחר הקלדת Return בחלון ה- entry העליון יתחלף תוכן חלון ה- text בתוכן הקובץ ששמו מצוין בחלון ה- entry (ההנחה היא שקובץ כזה קיים). בנוסף יעלם חלון ה- entry העליון, ומעל חלון ה- entry התחתון תופיע הכותרת "cut text from file" במקום הכותרת "cut text". לדוגמה, לאחר הקלדת Return בחלון ה- entry העליון במסך הקודם יופיע המסך הבא:

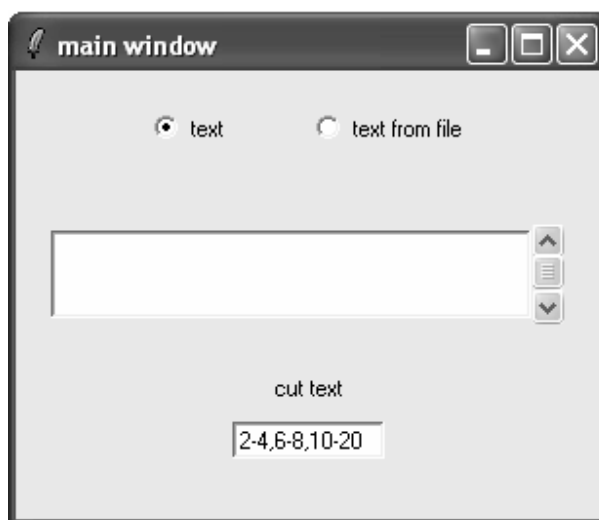


במסך זה חלון ה- text מציג את 3 השורות הראשונות של הקובץ F1. (חלון הגלילה שמימין לחלון ה- text מאפשר למשתמש להגיע לשורות האחרות שבקובץ F1).

ה- entry במסך זה פועל לפי אותם כללים שצוינו לעיל. דהיינו הקלדת Return בחלון ה- entry תגרום לכך שכל שורה שמופיעה בחלון ה- text תחתך לפי זוגות המספרים שצוינו בחלון ה- entry.

בחירת הכפתור (מסוג radiobutton) שנקרא: "text" במסך הקודם תגרום לכך שתוכן חלון ה-
text יתרוקן ומעל חלון ה-entry תופיע הכותרת "cut text" במקום "cut text from file".

לדוגמה, בחירת הכפתור text במסך הקודם תגרום להופעת המסך הבא:



בהצלחה !