

**סדנת תכנות ב - C**  
**תרגיל שני – מצביעים ומערכים**

תאריך הגשה: יום רביעי ה – 3.11.04

.1

**סעיף א (10%)**

נתון הקוד הבא (הקובץ q1a.c בתיקיית התרגיל)

```
#include <stdio.h>

int main() {
    int a = 7;
    int *p = &a, **p2 = &p, ***p3 = &p2;
    printf("%d %d %d %d %d %d\n",
        ***p3, (int)**p3,
        (int)(p3+5)-(int)p3,
        (p2+5)-p2,
        (a+5)-a,
        **p3 - p);
    return 0;
}
```

כתבו מה יהיה פלט התוכנית והסבירו אותו בליווי שרטוטים סכמטיים כדוגמת השרטוטים בהרצאות.

**סעיף ב (10%)**

בהתאם לקוד בסעיף הקודם, ציינו מה הטיפוס ומה המשמעות של הביטויים הבאים :

1. \*\*\*&p3
2. \*&p2
3. \*&p2
4. &\*\*p

לדוגמה: טיפוס הביטוי &p הוא int\*\* ומשמעותו היא הכתובת של המצביע p.

**סעיף ג (15%)**

נתון הקוד הבא (q1c.c):

```
#include <stdio.h>
#include <stdlib.h>
int main() {
    char *str = "The Road goes ever on and on \n Down from the door
where it began.\n Now far ahead the Road has gone,\n And I must
follow, if I can,\n Pursuing it with eager feet, \n Until it joins
some larger way \n Where many paths and errands meet. \n And whither
then? I cannot say. \n";

    int x = 7, y=2,z=19,c=0;
    size_t i,j,k;
    char ***ar3d = (char***)malloc(x*sizeof(char**));
    for(i=0; i<x; i++) {
        ar3d[i] = (char**)malloc(y*sizeof(void*));
        for(j=0; j<y; j++) {
            ar3d[i][j] = (char*)malloc(z*sizeof(char));
            for(k=0; k<z; k++)
                ar3d[i][j][k] = str[c++];
        }
    }

    /* ... */
}
```

}

עליכם להוסיף קוד במקום ה- /\*...\*/ / המדפיס את כל השיר המוצבע ע"י str (שיר מאת בילבו באג'ינס). הקוד שלכם אמור לקחת את כל האינפורמציה שלו רק מ- ar3d ולא מ- str.

הוסיפו קוד המשחרר את כל הזיכרון שהוקצע.

## 2. (25%)

כתבו תכנית המקבלת מספר ידוע של מילים, מכניסה אותם למערך של מחרוזות, ממיינת את המילים לפי סדר לקסיקוגרפי ומדפיסה את המילים לאחר המיון. קובץ ההרצה q2 שבתיקיית התרגיל מדגים את ההתנהגות הרצויה.

פונקציה ה- main שלכם צריכה להיות ממומשת באופן הבא (q2.c):

```
int main() {
    int n; /* number of words */
    char **table = readWords(&n);
    sort(table,n);
    print(table,n);
    freeTable(table,n);
    return 0;
}
```

ניתן להניח כי אורך מילה אינו עולה על 500 תווים. ניתן להשתמש במערך בגודל זה עבור קריאת המילים אך יש להקצות מערך בגודל מינימאלי עבור כל מילה המאוחסנת ב- table.

ניתן למיין את המילים ע"י אלגוריתם בעל סיבוכיות ריבועית. מימוש אלגוריתם בעל סיבוכיות nlogn – בונוס קטן.

## 3. (40%)

בשאלה זו תתבקשו לממש כלי שימושי מאד בשם grep. הכלי מאפשר לסנן קובץ ולהוציא ממנו שורות המכילות מחרוזת מסוימת. ניתן להפעיל את הכלי על מספר קבצים ואז הוא מבצע את הסינון על כל אחד מהם ומוסיף לתוצאת הסינון את שם הקובץ הרלוונטי. כמו כן ניתן להפעיל את הכלי על זרם הקלט הסטנדרטי. קובץ ההרצה q3 שבתיקיית התרגיל מדגים את ההתנהגות הנדרשת.

דוגמאות להפעלה:

```
$ q3 int check.c
```

ידפיס את כל השורות המכילות את המילה int בקובץ check.c.

```
$ q3 underhill *.txt
```

ידפיס את כל השורות בהם מופיעה המחרוזת underhill בכל קבצי הטקסט שבספרייה הנוכחית. בתחילת כל שורה יופיע שם הקובץ בו נמצאה השורה.

```
$ ls | q3 out
```

ידפיס את כל הקבצים המכילים את המחרוזת out.

בכל הדוגמאות הללו, החלפת q3 ב- grep יגרום לפלט זהה (נסו!). בכדי לקרוא על מאפיינים נוספים של grep השתמשו ב- man.

אין דרישה לממש את כל המאפיינים של grep רק את אלה הממומשים ב – q3.

ישנם מספר פרטים טכניים החסרים לכם על מנת לממש את q3. קובץ הדוגמה example.c שבתקיית התרגיל מדגים שימוש במאפייני השפה החסרים לכם. כל פונקציה שאינכם מכירים תוכלו לחפש ב – man. לדוגמה:

```
$ man fprintf
```

אינפורמציה לגבי עבודה עם קבצים חיצוניים ניתן למצוא גם ב – K&R עמוד 160.

**הגשה:**

קובץ tgz הכולל את הקבצים q1.c, q2.c, q3.c ו – readme. תדפיסים לתא הקורס.

תשובות לשאלות הדורשות כתיבה יש להוסיף כדף דף נפרד לתדפיסי התוכנית.

**בהצלחה**