

מבנה מחשב

תרגיל 4

אופטימיזציה

מועד אחרון להגשה: 28.1.2020

הנחיות הגשה:

- ההגשה ביחיד או בצוות של שני חברים.
- ההגשה באמצעות מערכת ההגשה בכתובת:

<http://submit.cs.biu.ac.il>

- אם ההגשה בצוות, כל אחד משני החברים יגישו את אותו קובץ למערכת ההגשה ויצינו בהערה בראש הקובץ את שמות שני החברים בצוות.

- ליצור ב-u2 תיקיה בשם opt:

```
mkdir opt
```

- להעתיק ממודל את הקובץ handout.tar לתיקיה opt.

- להיכנס לתיקיה:

```
cd opt
```

- לפרוס את הקובץ:

```
tar xvf handout.tar
```

- לערוך את הקובץ `kernels.c`:

```
team_t team = {
    "login1 login2",      /* Team name */
    "First1 Last1",       /* First member name */
    "first@nowhere.edu",  /* First member email */
    "First2 Last2",       /* Second member name */
    "second@nowhere.edu"  /* Second member email */
};
```

- לקרוא את ההערות בקובץ.

- לבנות את `driver`:

make

- להריץ את `driver`:

./driver

- להריץ את `driver -h`:

./driver -h

- לכתוב בקובץ `kernels.c` גרסאות יותר יעילות לפונקציות `rotate` ו-`smooth`,

לפי ההוראות שבתוך הקובץ.

- לבנות שוב את `driver`:

make

- לבדוק את הביצועים עם `driver`.

./driver

`driver` היא תכנית שיוצרת תמונה רנדומלית בגדלים שונים ובודקת את יעילות הפונקציות.

בקובץ `kernels.c` אפשר לרשום אליה כמה גרסאות של הפונקציות והיא תשווה ביניהם.

- להגיש רק את הקובץ `kernels.c`:

אפשר לבצע את הייעול והבדיקות בכל מחשב יוניקס, ההגשה תיבדק במחשב שהחומרה והתוכנה בו זהים ל-`u2`.

הציון מבוסס על הבדיקה של `driver`.

תיאור התרגיל

בתרגיל נייעל שתי פעולות בעיבוד תמונה :

rotate : פעולה שמסובבת תמונה ב- 90° נגד כיוון מחוגי השעון.

smooth : פעולה שמחליקה תמונה.

תמונה מיוצגת בדרך כלל על ידי מערך דו-ממדי של פיקסלים.

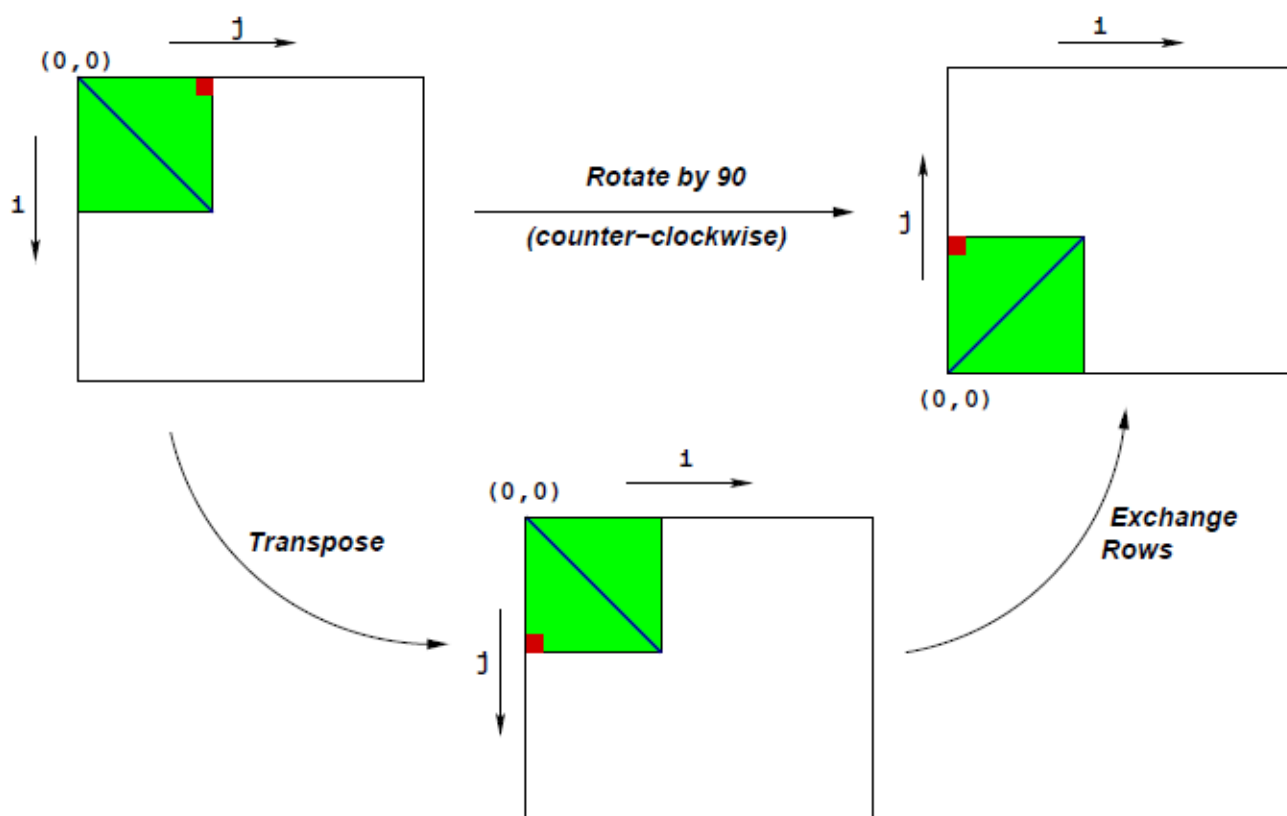
כל פיקסל הוא שלישייה של ערכי RGB.

בתרגיל כל התמונות ריבועיות.

rotate : הפעולה תמומש על ידי צירוף של שתי פעולות מטריצה :

Transpose: For each (i, j) pair, $M_{i,j}$ and $M_{j,i}$ are interchanged.

Exchange rows: Row i is exchanged with row $N - 1 - i$.

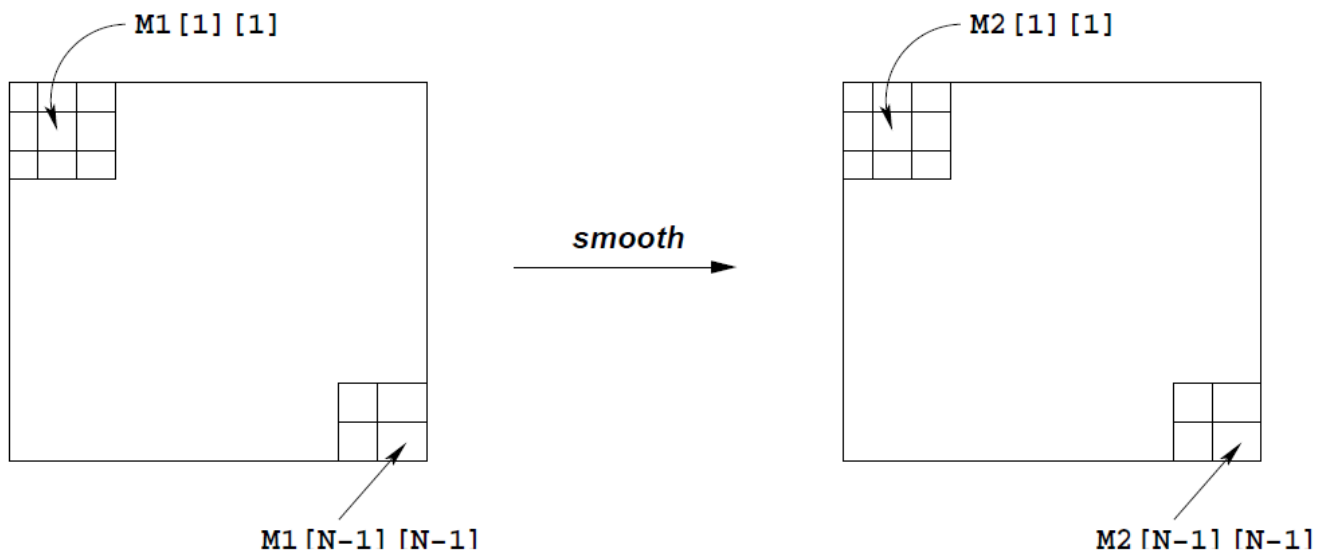


smooth : הפעולה תמומש על ידי החלפת הפיקסל (במטריצת היעד) בממוצע של ערכי הפיקסלים הסובבים אותו (במטריצת המקור).

הערכים של $M2[1][1]$ ושל $M2[N-1][N-1]$ בציור שלהלן הם :

$$M2[1][1] = \frac{\sum_{i=0}^2 \sum_{j=0}^2 M1[i][j]}{9}$$

$$M2[N-1][N-1] = \frac{\sum_{i=N-2}^{N-1} \sum_{j=N-2}^{N-1} M1[i][j]}{4}$$



מימוש

פיקסל מיוצג על ידי המבנה:

```
typedef struct {  
    unsigned short red; /* R value */  
    unsigned short green; /* G value */  
    unsigned short blue; /* B value */  
} pixel;
```

תמונה מיוצגת על ידי מערך חד ממדי של פיקסלים.

פיקסל i, j בתמונה שגודלה $n \times n$ נמצא במקום:

```
I[RIDX(i,j,n)]
```

RIDX מוגדר:

```
#define RIDX(i,j,n) ((i)*(n)+(j))
```

ראו `defs.h`.

מימוש נאיבי של `rotate` נמצא בקובץ `kernels.c`, יש ליעלו באמצעות טכניקות כמו:

`code motion, loop unrolling, blocking`

מימוש נאיבי של `smooth` נמצא בקובץ `kernels.c`, המימוש משתמש בפונקציה `avg` שבקובץ.

אין חובה להשתמש ב- `avg` במימוש היעיל, פה הייעול הוא יותר בזמן מאשר במקום.

בהצלחה