

תרגיל מס' 7 ישומי מחשב

מרצה פרופ' אלי פלקסר

הנושא: עיבוד תמונה וטרנספורמציות מרחביות של ווקטורים.

1. השתמש בדוגמא שנמצאת באתר הקורס, לביצוע מניפולציה על תמונה ששמורה בדיסק - בפורמט כל שהוא. על התוכנית לקרוא את התמונה לתוך מטריצה בזיכרון. על המטריצה הנ"ל יש לבצע את הפעולות הבאות:

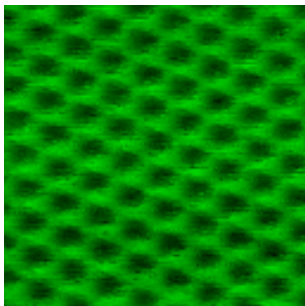
- הפעלת שיקוף סביב ציר X ושיקוף סביב ציר Y.
- הפעלת סיבוב ב-90 מעלות סביב המרכז.
- הפעלת סיבוב בזווית כל שהיא סביב נקודה כל שהיא בתמונה.

בכדי לבצע סיבוב השתמש [במטריצת סיבוב דו ממדית](#). זכור שהסיבוב מוגדר כסיבוב סביב הראשית.

2. במיקרוסקופים סורקים - כדוגמת [STM](#), [AFM](#) SNOM וכד', אינטרקציה פיסיקלית בין חוד למשטח, ממופה לתמונה תלת מימדית, בה צירי X ו Y מיצגים את המשטח הנבחן, וגווני הצבע מייצגים את התכונה הנמדדת בפני המשטח (גובה, צפיפות מטען, טמפרטורה ועוד). בתמונה למטה, מתוארת הדמייה של משטח גרפית בגודל 3 ננו מטר שהופק ע"י STM.

הורד מאתר הקורס את הקובץ [AfmGold.fep](#) ושמור אותו בכונן. הקובץ מכיל נתונים גולמיים שנימדדו ע"י מכשיר AFM. הקובץ הוא קובץ בינארי שמכיל 320 בתים של אינפורמציה כללית (בפורמט המוצג בדף הבא) שאחריהם מופיעים הנתונים של המדידות כמספרים שלמים מסוג short. כתוב תוכנית הקוראת את הקובץ ובונה ממנו תמונה טופוגרפית (מוצגת על CANVAS) בדומה לתמונה למטה. לאחר השימוש בפונקציה `load_pic()` יהיה לך מערך דו ממדי המכיל את נתוני המדידה בזכרון. השתמש בנתונים אלה בכדי ליצור את התמונה. שם לב שעוצמת הצבעים היא 8 ביט ולכן צריך לנרמל את הנתונים הגולמיים למרחב הצבעים. מומלץ ליצור נירמול סטנדרטי – כלומר נירמול של כל נקודה ביחס לסטיית התקן של התמונה כולה.

לאחר השלמת התמונה צור מנגנון בו אתה יכול לסמן קו ישר על התמונה, ולקבל בגרף את חתך הגבהים לאורך הקו.



```

/*=====*/
#include <ansi_c.h>

#define LENRES 82
#define PIC_SIZE_X 128
#define PIC_SIZE_Y 128

typedef struct {
    char id[4];
    int type;
    int pic_size_x;
    int pic_size_y;
    int size;
    int offsetx;
    int offsety;
    int mode;
    double voltage;
    double current;
    double specvol;
    int gain;
    int experiment;
    char time_date[30];
    char rem[80];
    int reserved[LENRES];
} PICHEAD;

/*=====*/
short vec[PIC_SIZE_Y][PIC_SIZE_X];
void load_pic()
{
    PICHEAD pich;
    FILE *fp;
    int i;

    fp=fopen("AfmGold.fep","rb");
    fread((char *)&pich, 1, sizeof(PICHEAD), fp);
    for (i=0; i<pich.pic_size_y; i++)
        fread(vec+i, sizeof(short), pich.pic_size_x, fp);

    fclose(fp);
}
/*=====*/

```