מחברת מס' ____ מתוך ____ מחברות	76634

אוניברסיטת תל-אביב

לפני התחלת הבחינה מלא את וקרא בעיון

תאריך הבחינה 22/6/03

שם הקורס תונע 1

שם המורה פרופ' בן הלפרין

החוג/המגמה מ"מ המחשב

הוראות

1. על הנבחן להיבחן רק בחדר שבו הוא רשום.
2. עם הכניסה לחדר הבחינה יש להניח את החפצים בצד לרבות מכשירי קשר ואמצעי תקשורת אחרים כשהם כבויים.
3. אסור להחזיק בהישג יד, בחדר הבחינה או בסמוך לו, כל חומר הקשור לבחינה/לקורס פרט לחומר שהשימוש בו הותר בכתב על ידי המורה.
4. יש למלא את הפרטים על מחברת הבחינה במקום המיועד לכך בלבד. אין לכתוב את השם או כל פרט מזהה אחר בתוך המחברת.
5. יש להישמע להוראות המשגיח. נבחן לא יעזוב את מקומו ללא קבלת רשות המשגיח. הפונה בשאלה או בבקשה ירים את ידו.

לשימוש המורה הבוחן:

הציון	100
המחברת נבדקה ביום	8/7/03
חתימת המורה	[illegible]

סמסטר ב' תשס"ג
מועד: א' 22/6/2003
משך הבחינה: 3 שעות
חומר עזר: שני דפי עזר

בחינה בקורס: **תוכנה 1**
מרצה: **פרופ' דן הלפרין**

הנחיות כלליות לבחינה:

- המבחן מורכב משלוש שאלות תכנות ושאלה אמריקאית אחת.
- חובה **לתעד** את שאלות התכנות.
- נא לכתוב בכתב קריא ובאופן מסודר.

- מספר סטודנט: ___

בהצלחה !

שאלה 1 (25 נקודות)

כתוב את הפונקציה char *letters (const char *str)
הפונקציה מקבלת מחרוזת ומחזירה מחרוזת המורכבת מן התווים שבמחרוזת המקורית, על פי סדר הופעתם, כך שאף תו אינו חוזר פעמיים. (שים לב שאין לדרוך על מחרוזת הקלט str .)

למשל, אם הפרמטר str הינו המחרוזת:

Hello everybody how are you?

הרי שהערך המוחזר יהיה המחרוזת:

Helo vrybdhwau?

שאלה 2 (30 נקודות)

כתוב פונקציה בשם create_matrix() שמקבלת תמונה המורכבת מפיקסלים, ואת מימדיה. הפונקציה מחזירה את התמונה בייצוג שונה, כמתואר להלן, או NULL במקרה של כשלון.

עליך להשתמש במבנים הבאים:

```
typedef unsigned char Pixel[3];

typedef struct element {
  struct element * left;
  struct element * top;
  struct element * right;
  struct element * bottom;
  Pixel pixel;
} Element;
```

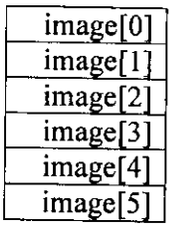
להלן ה- prototype של הפונקציה:

```
Element * create_matrix(Pixel * image, int width, int height);
```

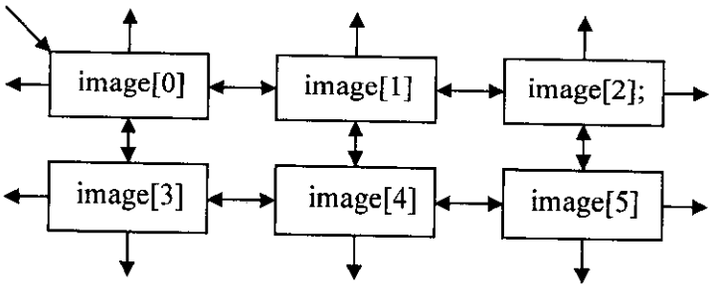
הפונקציה מקבלת מצביע למערך חד מימדי בגודל width * height של אברים מסוג Pixel.
השורות מופיעות בסדר עוקב בזיכרון. כל שורה היא בגודל width, וישנן height שורות.

הפונקציה מחזירה מצביע למבנה של רשימה מקושרת "דו-מימדית" המורכבת מרשומות מסוג Element.
כל Element מכיל Pixel, ו-4 מצביעים ל-4 רשומות מסוג Element לכל היותר אשר מכילים את הפיקסלים משמאל, מלמעלה, מימין ומלמטה בהתאמה, אם קיימים, כפי שמתואר בציור. אם פיקסל לא קיים, המצביע המתאים הוא NULL.

לדוגמא, אם width = 3 ו- height = 2, אזי הקלט ייראה כך:



והתוצאה תהיה:



שאלה 3 (35 נקודות)

נתון שלד של תכנית הקוראת מתוך קובץ בינארי מספר שלם n ו- $2n^2$ מספרי double שהם אברי שתי מטריצות ריבועיות A ו-B בגודל nxn כל אחת. התכנית מחשבת את מטריצת המכפלה C תוך שימוש בתהליך ילד נפרד עבור חישוב של כל איבר ב-C והעברת האינפורמציה לתהליך ההורה דרך pipe.

א. (15 נקודות)

כתוב הפונקציות allocate_square_matrix() (שמחזירה NULL בכשלון) ו- get_input() (שמחזירה 0 בסיום תקין ו- 1- בכשלון).

אברי המטריצה A מופיעים בקובץ ברצף שורת מטריצה אחרי שורה ואחריהם באופן דומה מופיעים אברי B.

אין צורך לכתוב את פונקציות העזר הנוספות. ניתן להניח שהן כתובות ועומדות לרשותנו. בפרט הפונקציה multiply_ij() המחשבת את האבר ה-[i][j] במטריצת התוצאה.

ב. (20 נקודות)

השלם את ה-main(). ניתן להוסיף טיפוסים ומשתנים מעבר לאלו המופיעים בשלד.

```
/* use pipes to multiply two nxn matrices */
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>

typedef double **matrix;

matrix allocate_square_matrix(int dimension);
int    get_input(char *file_name, int *dimension, matrix *A, matrix *B);
double multiply_ij(matrix A, matrix B, int dimension, int i, int j);
void   print_matrix(char *header, matrix A, int dimension);
void   error_exit(char *s);

int main(int argc, char **argv)
{
  matrix A, B, C;
  int i, j;
  int n;              /* square matrix dimension */
  int pd[2];          /* pipe descriptor */

  if (get_input(argv[1], &n, &A, &B) == -1)
    erro_exit("get_input() failed");

  if (pipe(ad) < 0)  /* create a pipe */
    error_exit("pipe() failed");

  /* complete this part of the main,
   * you may add more typedefs and variables
   */

  print_matrix("the matrix C", C, n);
  return 0;
}

matrix allocate_square_matrix(int dimension)
{
  /* complete */
}

int get_input(char *file_name, int *dimension, matrix *A, matrix *B)
{
  /* complete */
}
```

שאלה 4 (10 נקודות)

הקטע הבא הוא ה-makefile של תרגיל 6:

```
SPECFILE ?=spec.t
gen: gen.o
asm: asm.o inst.o
sim: sim.o
inst.o: inst.h
asm.o: inst.h
sim.o: inst.h
inst.h inst.c: $(SPECFILE) gen
      gen $<
%.e : %.s asm
      asm $<
clean:
      rm -rf gen.o inst.o asm.o sim.o gen asm sim inst.c inst.h *.e
```

כשכותבים " ls –clt " (פקודה המדפיסה את תוכן ה- directory בסדר יורד של זמן עידכון אחרון), מקבלים:

```
-rw-r--r--    1 efif     math          167 May 23 22:47 spec.t
-rwxr-xr-x    1 efif     math          21k May 23 22:45 sim*
-rw-r--r--    1 efif     math         9.6k May 23 22:45 sim.o
-rw-r--r--    1 efif     math           44 May 23 22:45 prog.e
-rwxr-xr-x    1 efif     math          23k May 23 22:45 asm*
-rw-r--r--    1 efif     math          11k May 23 22:45 asm.o
-rw-r--r--    1 efif     math         1.8k May 23 22:45 inst.o
-rw-r--r--    1 efif     math          179 May 23 22:45 inst.c
-rw-r--r--    1 efif     math          618 May 23 22:45 inst.h
-rwxr-xr-x    1 efif     math          25k May 23 22:34 gen*
-rw-r--r--    1 efif     math          14k May 23 22:34 gen.o
-rw-------    1 efif     math         2.3k May 23 22:24 asm.c
-rw-------    1 efif     math         3.6k May 23 22:24 gen.c
-rw-------    1 efif     math          247 May 23 22:24 makefile
-rw-------    1 efif     math         1.2k May 23 22:24 sim.c
-rw-r--r--    1 efif     math           30 May 10 10:54 prog.s
```

מה נקבל כשנכתוב "make prog.e sim" ? בחר תשובה אחת מהבאות ותן הסבר קצר לבחירה (במחברת הבחינה).

1. gcc -g -c -o gen.o gen.c gcc -g gen.o -o gen gen spec.t gcc -g -c -o asm.o asm.c gcc -g -c -o inst.o inst.c gcc -g asm.o inst.o -o asm asm prog.s gcc -g -c -o sim.o sim.c gcc -g sim.o -o sim	3. gcc -g -c -o asm.o asm.c gcc -g -c -o inst.o inst.c gcc -g asm.o inst.o -o asm asm prog.s gcc -g -c -o sim.o sim.c gcc -g sim.o -o sim
2. gen spec.t gcc -g -c -o asm.o asm.c gcc -g -c -o inst.o inst.c gcc -g asm.o inst.o -o asm asm prog.s gen spec.t gcc -g -c -o sim.o sim.c gcc -g sim.o -o sim	4. gen spec.t gcc -g -c -o asm.o asm.c gcc -g -c -o inst.o inst.c gcc -g asm.o inst.o -o asm asm prog.s gcc -g -c -o sim.o sim.c gcc -g sim.o -o sim

```
char *letters(const char *str)

    int i; /* loop counter */
    char asc[256]; /* indication which charecters we already got */
    int len; /* the input string length */
    char *ret; /* the return string */
    int ret_len; /* the length of the return value */

    len = strlen(str);

    ret = (char *) malloc ((len+1) * sizeof(char)); /* allocating memory
    if (ret==NULL)                                    and checking
        return NULL;                                  pointer */

    ret_len=0;

    for (i=0; i<256; i++) {          /* initializing the "asc"
        asc[i] = 0;                     array to be all
    }                                   zero's */

    for (i=0; i<len; i++) {          /* main loop:
        if (asc[str[i]] == 0) {         running over the
            ret[ret_len] = str[i];      input string, if
            asc[str[i]] = 1;            we encounter a
            ret_len++;                  "new" charecter
        }                               we add it
    }                                   to the output
                                        string and
                                        change it's value
                                        in the asc array
                                        to be 1 */

    ret[ret_len] = '\0';   /* adding the \0 at the end of the
                              output string */
    return ret;
}
```

הערות:

1. הבנתי שמשתמשים במערך asc יכול לסמן לו אותיות כבר הופיעו בכלל ואילו עדיין לא.
2. שים הבחנה יפה מתוכנה אשר ב לשחרר בעזרת free הינו תגית נאורים המשתנים אם מקבלים צריכים להיות לתוצאה ה- לצורך בעזרת realloc(ret,ret_len+1) אם רוצים.
3. בעקרון אפשר חוסר בזכרון ואז מחזיר קלט יוקב בפונקציה מחזיר NULL

30/30 שאלה 2

```
Element *create_matrix (Pixel *image, int width, int height) {

    int i;  /* loop counter */
    Element *ret; /* the return pointer */
    int len; /* the length of the image */
    assert (width>0);
    assert (height>0);
    len = width*height;          /* setting the length of the array */

    ret= (Element *) malloc (len * sizeof (Element)); } /* memory allocation
    if (ret == NULL)                                    for the new
        return NULL;                                    picture format */


    for (i=0; i<len; i++) {                  /* running over all the pixels */

        ret[i].pixel[0] = image[i][0];
        ret[i].pixel[1] = image[i][1];       } /* setting the pixels
        ret[i].pixel[2] = image[i][2];          RGB values */

        if ((i%width)==0)
            ret[i].left = NULL;              } /* if left pixel exists
        else                                    set the left pointer
            ret[i].left = &(ret[i-1]);          to point it, if
                                                not set it to
                                                be null */

        if ((i%width)==(width-1))
            ret[i].right = NULL;             } same for right
        else                                    pointer
            ret[i].right = &(ret[i+1]);

        if ((i/width)==0)
            ret[i].top=NULL;                 } Same for top
        else                                    pointer
            ret[i].top=&(ret[i-width]);

        if ((i/width)==(height-1))
            ret[i].bottom = NULL;            } same for
        else                                    bottom
            ret[i].bottom = &(ret[i+width]);    pointer

    }

    return ret;
}
```

הערות:

1. הפונקציה תגיר את הרשימה כפי שראשה הוא מוקצה בבלוק אחד באורך len*sizeof(Element) בזיכרון.
2. הפונקציה תגיר NULL במקרה ואין זיכרון.
3. זכור תמונה בו ריקה (גודל 0x0) הפונקציה תשלח הודעה שאין לה שום

15/15 עמ' 3

7/7

```
matrix allocate-square-matrix(int dimension){
    int i;          /* loop variable */
    matrix ret;     /* return value */

    ret=(matrix)malloc(dimension * sizeof(double *));
    if (ret==NULL)
        return NULL;

    for (i=0; i<dimension; i++){
        ret[i]=(double *)malloc(dimension * sizeof(double));
        if (ret[i]==NULL){
            for (i--; i>0; i--){
                free(ret[i]);
            }
            free(ret);
            return NULL;
        }

    }

    return ret;
}
```

/* allocating the main pointers pointer */

/* allocating a doubles array in each cell

/* if we encountered error in allocation freeing all the already allocated data and return NULL */

8/8

```
int get_input(char *file_name, int *dimension, matrix *A, matrix *B){
    FILE *fp;   /* input file pointer */
    int i;      /* loop variable */

    fp=fopen(filename,"r");
    if (fp==NULL)
        return -1;

    if (fread(dimension, sizeof(int),1, fp) != 1){
        fclose(fp);
        return -1;
    }
    A=&(Allocate_square_matrix(*dimension));
    B=&(Allocate_square_matrix(*dimension));
    if ((A==NULL) || (B==NULL)){
        fclose(fp); return -1;
    }
    for (i=0; i<(*dimension); i++){
        if (fread((*A)[i], sizeof(double), *dimension, fp) !=(*dimension)){
            fclose(fp); return -1;
        }}
    for (i=0; i<(*dimension); i++){
        if(fread((*B)[i], sizof(double), *dimension, fp) != (*dimension)){
            fclose(fp); return -1;
        }}
    fclose(fp);                 /* closing the file */
    return 0;
```

פתיחת הקובץ ובדיקת FILE* ב תקינות

dimension קריאת

קריאת מטריצה A
+
קריאת מטריצה B

קריאת שורות

הערות לשאלה 3 מתוך 5

15/20

```
int pid;
int x, y;
double tmp;
```

} חסרים משתנים

קוד הקיר שיש לתקן:

```
for (i=0; i<n; i++) {

    for (j=0; j<n; j++) {
        pid = fork();
        if (pid==0)
        {
            if(write(pd[1], &i, sizeof(int)) != sizeof(int))
                error_exit ("error in pipe");

            if(write(pd[1], &j, sizeof(int)) != sizeof(int))
                error_exit ("error in pipe");
            tmp = multiply_ij(A,B,n,i,j);
            if(write (pd[1], &tmp,
                    sizeof(double)) != sizeof(double))
                error_exit ("error in pipe");

            return 0;
        }
    }
}

for (i=0; i<n; i++) {
    for (j=0; j<n; j++) {
        if (read(pd[0], &x, sizeof(int)) != sizeof(int))
            error_exit ("error in pipe");

        if(read(pd[0], &y, sizeof(int)) != sizeof(int))
            error_exit ("error in pipe");

        if (read (pd[0], &(C[x][y]), sizeof (double)) != sizeof(double))
            error_exit ("error in pipe");
    }
    close (pd[0]);
    close (pd[1]);
}
```

} יותר מדי כמספר הפעמים שקוראים.

} סגירת הפיפים.

לא יעבוד
לא לגיר
write
אחר נמצאים
לתוך הפיפים
לכן יש לשנות
את 3 הראשונים
בלולאה
לתוך
write
אחרי

-5

הערות:

1. כל process מאבד את הקואורדינטות i,j ואולם ה-pipe את i,j ותוצאת הכפלה.

2. בסוף פרוסה האב קוראים את הערכים (קואורדינטות ותוצאה) ומכניס למערך C הקואורדינטות המתאימות את התוצאה

שאלה 4 $\frac{10}{10}$

הרשומה הראשונה היא 4.

4 הסבר:

בהקצאה gen.spec לא ניסה להרחיב ולכן הרשומה הראשונה אינה 3.

מכיוון שהקצאה gen.spec לא הפעילה פעולים באותה דרגה של make, הרשומה הראשונה לא יכולה להיות 3.

gen.o תמיד יותר מ gen.c ולכן אין צורך לבדוק אותו ולכן רשומה 1 אינה נכונה.