

הוראות לנבחנים ולנבחנות (נכתבו בלשון זכר אך נועדו לשני המינים) לפני התחלת הבחינה מלא את כל הפרטים הבאים בכתב ברור וקרא בעיון את ההוראות:

1. הנך נדרש לשמור על טוהר הבחינה ועל עבודה עצמית ולהישמע להוראות המשיגים ולנהלי האוניברסיטה. אין להעתיק, אין לדבר ואין להעביר חומר בין הנבחנים.

נבחן הנוהג בניגוד להוראות צפוי להפסקת בחינתו ולהעמדה לדין משמעתי.

2. על הנבחן להבחן בחדר שבו הוא רשום.

3. אין להחזיק **טלפונים ניידים** או אמצעי תקשורת ומכשירים אלקטרוניים כלשהם בזמן הבחינה. על הנבחן להניח את כל חפציו האישיים בצד החדר הרחק ממקום מושבו.

4. אין להחזיק בהישג יד, בחדר הבחינה או בסמוך לו, כל חומר הקשור לבחינה או לקורס פרט לחומר שהשימוש בו הותר בכתב על ידי המורה.

5. קריאת השאלון מותרת רק לאחר קבלת רשות מהמשיג.

6. נבחן לא יעזוב את מקומו ולא את חדר הבחינה בטרם סיים את הבחינה ללא קבלת רשות מהמשיג. בעת יציאה מן החדר, יפקיד הנבחן את מחברות הבחינה והשאלון (טופס הבחינה) בידי המשיג.

7. נבחן שנכנס לחדר הבחינה וקיבל את השאלון לידיו, לא יהא רשאי לעזוב אותו אלא כעבור חצי שעה לפחות ממועד תחילתה ורק לאחר שיחזיר למשיג את המחברת ואת השאלון, ויקבל ממנו את התעודה המזהה שאותה מסר עם כניסתו לכיתה. נבחן שהחליט לעזוב בלי לכתוב את הבחינה ייחשב כמי שנבחן במועד זה וציונו יהיה "0".

8. אין לכתוב את השם או כל פרט מזהה אחר בתוך המחברת. פרטי הנבחן ימולאו על כריכת המחברת במקום המיועד לכך בלבד.

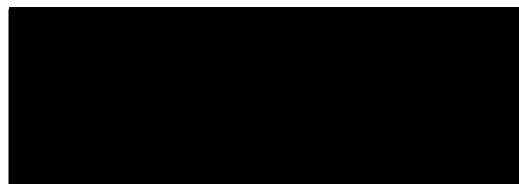
9. אין לחלוש דפים מהמחברת. טיוטה תיכתב בתוך המחברת בלבד. אין להשתמש בדפים שהביא הנבחן.

10. יש לכתוב את התשובות בעט כחול או שחור, בכתב יד ברור ונקי. בתום הבחינה יחזיר הנבחן את המחברת והשאלון ויקבל מיד המשיג את התעודה המזהה.

11. אין לכתוב מעבר לקו האדום משני צידי הדף.

בהצלחה.

תאריך הבחינה 28.1.07
שם הקורס סכ"ק תוכנה
שם המורה פרופ' רפי אלון
החוב/המגמה אצ"מ



לשימוש המורה הבוחן:

הציון 108
המחברת נבדקה ביום _____
חתימת המורה _____

102825

סיכום

	<u>global</u>	<u>main</u>	<u>f</u>
a	5	1	
b		2 1 1	
c		2 1	

קודם 2 מחזורי
 בקריאה ואחריו מ
 מילוי עמודי הביוק.
 מופיעים קוד רוחים!
 על היכנסים מסתב
 אנסיוורס, וזה צפוי!
 עם פטרים!

Before: 1, 2, 3

After:

temp = fra

f: q = ~~a~~ k
 p = b k
 tmp = b k

f: q = b k
 p = c k
 tmp = c k

THE UNIVERSITY OF CHICAGO

אורבוסטוטת תל-אביב

משרד החינוך והשכלה גבוהה

$$3 \quad 3 \quad x = 00ef =$$

3 3 3

$$\boxed{x = 29}$$

५७७१०१

5 2 1

50104

↓ ↓ ↓ ↓ ↓ ↓ ↓


$$4 \times 2 \times 4 = 32$$

↓ ↓

~~Handwritten signature~~

שבת שלום רב
לכל המשפחה.

0000 0000 0000 0000



1. file name

```
#include <stdio.h>
```

```
/* Here comes all the "includes" needed for  
the program */
```

```
int main(int argc, char **argv) {
```

```
    /* Variable declarations. */
```

```
    int N = 0, i
```

```
    FILE *input, *output;
```

```
    char **before, *word, *
```

```
    char[] = word[10];
```

```
input = fopen  
fscanf
```

```
    if (argc != 3) // Wrong input parameters.  
        return -1;
```

```
    input = fopen(argv[1], "r"); assert(input != NULL);
```

```
fscanf fscanf(input, "%d", &N);
```

```
    before = malloc(N * sizeof(char) * 10);
```

```
for (i = 0; i < N; i++) { // Scans the strings  
    word = malloc(10 * sizeof(char);  
    into the array;
```

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

```
        fscanf(input, "%s", word);
```

```
        before[i] = (char *) word;
```

```
    }
```

```
    fclose(input);
```

```
    qsort(before, N, sizeof(char) * 10, compare_str);
```

```
    word = realloc(1 * sizeof(char);  
    tmp = str;
```

```

for (i=0
output = fopen ("argu[2]", "w");
for (i=0; i < N; i++) {
tmp = str;
word = strcat (before[i], "N");
printf (output, "%s", word);
}
fclose (output);
free (before);
free (word);
free (input);
free (output);
input = NULL; output = NULL; word, before = NULL;

```

```

int compare_str (char *a, char *b);
int is_a_bigger = 1;
return

```

```

int compare_str (const void *a, const void *b) {
char *a1, *b1;
a1 = malloc (sizeof(char) * 10 + 1);
b1 = malloc (sizeof(char) * 10 + 1);

```

```

a1 = strdup ((char *) a);
b1 = strdup ((char *) b);
if (strcmp (a1, b1) == 0) return 0;

```

~~for~~

ch = a[i];

while (i < 10) {

if (ch > b[i]) {

~~return a - bigger = 1;~~ return 1;
break;

}

if (b[i] > 'ch') {

~~return~~
~~a - bigger = -1;~~

~~break;~~

if (b[i] == '\0') return -1;

if (a[i] == '\0') return -1;

i++;

~~ch = a[i];~~

}

:1 אלף דולר

#include <stdio.h>

/* there comes all the includes in the program */

int compare_str (const void *a, const void *b) {

int i = 0;

char *a1, *b1;

a1 = malloc (10 * sizeof(char) + 1);

b1 = malloc (10 * sizeof(char) + 1);

a1 = strdup((char *) a);

b1 = strdup((char *) b);

/* Now a1 and b1 contains the strings for
comparation method */

if (strcmp(a1, b1) == 0) return 0;

/* We know now that they are not equal.

while (a[i] != '\0') {

if ((char) a[i] > (char) b[i])
return 1;

if ((char) b[i] > (char) a[i])
return -1;

if (b[i] == '\0')
return 1;

i++;

} //End while;

if (b[i] != '\0') return -1;

else return 1;

} //End compare_str;

#include <stdio.h>

-2

int compare_str(const void *a, const void *b) {

int i = 0;

char *a1, *b1;

a1 = malloc(10 * sizeof(char) + 1);

b1 = malloc(10 * sizeof(char) + 1);

a1 = strdup(char *) a;

b1 = strdup(char *) b;

/* Now a1 and b1 contains the strings */

if (strcmp(a1, b1) == 0) {

free(a1);

free(b1);

return 0;

/* Now we know that they are not equal
while (a[i] != '\0') {

if ((char) a[i] > (char) b[i]) {

free(a1); free(b1); return 0;

if ((char) b[i] > (char) a[i]) {

free(a1); free(b1); return -1;

if (b[i] == '\0') {

free(a1); free(b1); return 1;

} // End while;

free(a1); free(b1);

return -1; // Because a[i] = '\0'

} // End Compare_str


```
int main (argc, char **argv) {  
    //Variable Declarations:
```

```
    int N=0, i=0;  
    FILE *input, *output  
    char **words, *word;  
    char ch;
```

```
    assert (argc == 3);  
    input = fopen(argv[1], "r");  
    assert (input != NULL);  
    fscanf (input, "%d", &N);  
    word = malloc (1011 * sizeof(char) + 1); word = "10";  
    words = malloc (N * sizeof sizeof(word));  
    fscanf(inputinput, "%c", &ch); //scans the first blank.  
    fscanf(input, "%c", &ch);  
    while (ch != EOF) {  
        if (!is space(ch)) {  
            words[i] = word; fscanf(input, "%c", &ch);  
            i++; }  
        else {  
            word = strcat(word, ch);  
            fscanf(input, "%c", &ch);  
            i++; }  
    } //End while.
```

/*We now scanned all the characters from
the file to an array, now lets sort it!*/
fclose(input);

```

qsort(words, N, sizeof(word), compare_str);
// Now words array is LEX sorted.
// Lets write him to a file!
word = realloc(0 + 10 * sizeof(char); word = '\0';
output = fopen(argv[2], "w"); assert(output != NULL);
for (i = 0; i < N; i++) {
    word = strcat(words[i], ' ');
    fprintf(output, "%s", word);
}

```

```

word = strcat(words[i]);
fprintf(output, "%s", word);

```

/* Free all allocated memory and close the files: */

```

fclose(output);
free(words);
free(word);

```

```

return 0;
} // End main.

```

ת.ז. 040877938
מ.ס. מחברת 42

סמסטר א' תשס"ז, מועד א'

28/1/2007

משך הבחינה: שעתיים
חומר עזר: שני דפי עזר

בחינה בקורס: פרויקט תוכנה
מרצים: פרופ' דניאל כהן-אור וד"ר רוזד שרן

הנחיות כלליות לבחינה:

- בראש העמוד הראשון של טופס המבחן (עמוד זה) יש לציין את מספר תעודת הזהות.
- בבחינה חמש שאלות (פתוחות ואמריקאיות) בעלות ניקוד משתנה, בסך של 110 נק'.
- את התשובות לשאלות האמריקאיות יש למלא בטבלה המיועדת לכך (בעמוד זה).
- חובה לתעד את התשובות לשאלה הפתוחה (כמובן מותר בעברית).
- בתשובה לשאלה הפתוחה יש לכלול את כל ההכרזות הדרושות, אך אין צורך להוסיף לקטעי הקוד פקודות #include.

בהצלחה !

טבלת תשובות לשאלות האמריקאיות

שאלה 2	א
שאלה 3	ה
שאלה 4	ז
שאלה 5	3

שאלה מס' 1 (50 נק')

כתוב/כתבי תכנית המקבלת כארגומנטים ב-command line שתי מחרוזות המייצגות שמות קבצי טקסט: קובץ קלט וקובץ פלט. קובץ הקלט מכיל מספר N חיובי ואחריו N מלים בנות עד 10 תווים אלפבתיים כל אחת. המספר והמלים מופרדים זה מזה ע"י רווחים. על התכנית לסדר את המלים הנ"ל בסדר לקסיקוגרפי, מקטנה לגדולה, ולכתוב אותן בסדר הזה בקובץ הפלט, מופרדות בפסיקים. לדוגמא, הקובץ שבו "3 cd aab tt" יביא ליצירת הפלט: "aab,cd,tt".

שאלה מס' 2 (15 נק')

נתון קטע הקוד הבא (הנח קיום כל ההכרזות הדרושות):

```
void f(int *q, int * p)
{
    int *temp = p;
    *p = *q;
    q = temp;
}

int a = 5;

int main()
{
    int a = 1, b = 2, c = 3;

    printf("Before: %d,%d,%d\n", a, b, c);
    f(&a, &b);
    f(&b, &c);
    printf("After : %d,%d,%d\n", a, b, c);
    return 0;
}
```

מה יהיה פלט התוכנית (משמאל לימין)?

א. התוכנית תדפיס:

Before: 1,2,3

After : 1,1,1

ב. התוכנית תדפיס:

Before: 1,2,3

After : 2,3,1

ג. התוכנית תדפיס:

Before: 5,2,3

After : 1,1,1

ד. התוכנית תדפיס:

Before: 5,2,3

After : 2,3,1

שאלה מס' 3 (15 נק')

נתון קטע הקוד הבא (הנח קיום כל ההכרזות הדרושות):

```
int f(unsigned short x)
{
    int count;

    for (count=0; x!=0; x>>=1) {
        if (x & 1)
            count++;
    }
    return count;
}

int main(void)
{
    unsigned short x = 0x00ef;

    while (x) {
        printf("%d ",f(f(x)));
        x <<= 4;
    }
    printf("\n");
    return 0;
}
```

מה יהיה פלט התכנית (משמאל לימין)?

א. 3 3 1 0
ב. 3 3 3 1
ג. 7 7 4 0
ד. 7 7 7 4

שאלה מס' 4 (15 נק')

נתון קטע הקוד הבא (הנח קיום כל ההכרזות הדרושות):

```
int main (void){
char student[64] = "Moshe Cohen";
char *teacher;

X/* line 1 */ strcpy(teacher, student);
/* line 2 */ teacher = student;
/* line 3 */ strcpy(student, "Albert Einstein");
X/* line 4 */ student = teacher;
    return 0;
}
```

אילו מהשורות הבאות תקינות:

א. כל השורות

ב. רק 1, 2 ו-4

ג. רק 2 ו-3

ד. רק 2, 3 ו-4

שאלה מס' 5 (15 נק')

נתון קטע קוד הבא (הנח קיום כל ההכרזות הדרושות):

```
int main(void)
{
    char c, ch = 0;
    int counter = 1;

    if (scanf("%c", &c) != 1) {
        return -1;
    }
    while (ch != '\n') {
        scanf("%c", &ch);
        if (ch == c)
            counter++;
    }
    printf("%d\n", counter);
    return 0;
}
```

איזו מהטענות הבאות נכונה:

א. התכנית קוראת מספר נתון של תווים מהקלט.

ב. התכנית סופרת את מספר התווים בשורה.

ג. התכנית סופרת את מספר ההופעות של תו מסוים בשורה.

ד. התכנית מכילה שגיאה.

בהצלחה!