

11

סמסטר א' תשס"ה
מועד: א' 06/02/2005
משך הבחינה: 3 שעות
חומר עזר: שני דפי עזר

בחינה בקורס: תוכנה 1
פרופ' דניאל כהן-אור

הנחיות כלליות לבחינה:

- המבחן מורכב מארבע שאלות תכנות. יש להשיב על כל השאלות. שימו לב כי סכום כל השאלות 112
- חובה לתעד כל פעולה לא טריוויאלית שנעשית.
- יש לכתוב קוד קריא ויעיל ככל האפשר.
- נא לכתוב בכתב קריא ולא מחובר.

בהצלחה !

שאלה מס' 1 (46 נקודות)

נתוני גן ילדים מוחזקים בשני מבני נתונים שונים. נתוני הילדים מוחזקים בתוך עץ בינארי, הממוין לפי שם המשפחה של כל ילד. נתוני הורים מוחזקים בתוך רשימה מקושרת, הממוינת לפי משכורתם הממוצעת של שני ההורים (סדר יורד). עבור כל ילד מוחזק גם שמו הפרטי, גילו וסדר ישיבתו בכיתה (1 עד n). לכל זוג הורים מוחזק גם מצבם המשפחתי (גרוש=1, נשוי=0)

צומת בעץ הילדים מתואר על ידי מבנה הנתונים הבא:

```
typedef struct node {
    struct node *left;
    struct node *right;
    char    *sur_name;
    char    *name;
    float   age;
    short   order_in_class;
} Tree_Node;
```

איבר ברשימה המקושרת של ההורים מתואר על ידי מבנה הנתונים הבא:

```
typedef struct node {
    struct node *next;
    float   avg_salary;
    char    *sur_name;
    bool    is_divorced;
} List_Node;
```

בשאלות הבאות ניתן להניח כי קיימת התאמה חד-חד ערכית ועל בין הילדים להוריהם בשני מבני הנתונים. לפיכך, לכל ילד בעץ קיים זוג הורים תואם אחד ויחיד ולהיפך (מבחינת שם משפחה sur_name). כמו כן לא ניתן להניח כי הפרמטרים הנשלחים לפונקציה לא ימחקו במהלך הריצה של התוכנית (אלא אם צוין אחרת). לא ידוע מספר הילדים n מראש.

א. (18 נקודות) כתוב פונקציה אשר מקבלת כקלט את שני מבני הנתונים ומחשבת את המיזוג של שניהם. נתוני המיזוג בין שני מבני הנתונים יוחזרו בתוך מערך ממיון (כתובת וגודל) לפי סדר ישיבתם של התלמידים בכיתה (סדר עולה). חתימת הפונקציה (prototype) היא:

```
void Merge(Tree_Node *tree_root, List_Node *list_head, Merge_Cell
**array, int *array_size);
```

תא במערך המיזוג מתואר על ידי מבנה הנתונים הבא:

```
typedef struct merge_cell {
    char    *sur_name;
    char    *name;
    float   age;
    short   order_in_class;
    float   avg_salary;
    bool    is_divorced;
} Merge_Cell;
```

ב. (18 נקודות) כתוב פונקציה אשר מקבלת כקלט את שני מבני הנתונים ומחשבת את תת-קבוצת כל הילדים אשר הוריהם גרושים ומשכורתם הממוצעת עולה על 5000. שמות ושמות משפחת הילדים (בלכד) יוחזרו בתוך רשימה מקושרת (פוינטר לראשה) אשר ממוינת לפי משכורת ההורים הממוצעת (סדר יורד). על הפונקציה להיות יעילה ככל שניתן מבחינת זמן ריצה). חתימת הפונקציה (prototype) היא:

```
Select_Cell* Select(Tree_Node *tree_root, List_Node *list_head);
```

איבר ברשימה המקושרת המוחזרת מתואר על ידי מבנה הנתונים הבא:

```
typedef struct select_cell {
    struct select_cell *next;
    char *sur_name;
    char *name;
} Select_Cell;
```

ג. (10 נקודות) רוצים לאפשר שינוי גילו של ילד בצורה מהירה. הצע שינוי במבנה תא המערך (struct) אשר בסעיף א. כך שאפשר יהיה לעדכן את גילו של ילד במערך מבלי לעדכן אותו גם בעץ. הנח כי שני המבנים לא נמחקים במהלך התוכנית.

שאלה מס' 2 (22 נקודות)

כתוב תוכנית (main) אשר מחשבת עבור שתי רשימות מספרים אילו מבין הזוגות הסדורים (אחד מכל רשימה) נמצאים במרחק ϵ (Epsilon) זה מזה. התוכנית מקבלת ארבעה פרמטרים. שלושה שמות קבצים ופרמטר ϵ . בקובץ הראשון מבין השניים, שמורים מספרים בפורמט טקסט (ascii) בדיוק של שלמים (int). בקובץ השני מבין השניים שמורים מספרים בפורמט בינארי בדיוק של (float). הקובץ השלישי נועד לשמור את התוצאה. הפרמטר הרביעי לתוכנית הוא מספר ϵ בדיוק של (float). על התוכנית לבדוק האם ההפרש (בדיוק float) בין שני מספרים סדורים (האחד מכל רשימה-קובץ), הוא קטן שווה ϵ . עבור כל זוג אשר מקיים את התנאי, על התוכנית לרשום את ההפרש המתקבל, בפורמט בינארי בקובץ השלישי.

הערות:

הינך יכול להניח כי קיים אותו מספר של מספרים בשני קבצי הקלט.
בקובץ הטקסטואלי, ניתן להניח כי בכל שורה חדשה קיים מספר אחד ויחיד המסתיים בתו '\n'.
לא ניתן להניח כי קבצי הקלט קיימים תמיד.

שאלה מס' 3 (22 נקודות)

נגדיר את הטיפוס הבא (צומת ברשימה מקושרת):

```
typedef struct _Word{
    char *str;
    struct _Word *next;
}Word;
```

כתוב פונקציה בעלת ה-prototype הבא:

```
Word* WordParser (char *str, char **separators, int size);
```

אשר מקבלת כקלט מחרוזת (str) מערך של מחרוזות מפרידות (separators) וגודלו של המערך (size), ומחלקת את המחרוזת הנתונה לתתי-מחרוזות תוך שימוש במערך המחרוזות המפרידות כנקודות החלוקה.
תתי-המחרוזות יוחזרו ברשימה מקושרת (מסודרות לפי סדר הופעתן במחרוזת המקורית), כאשר כל צומת ברשימה מכיל מצביע לתתי-מחרוזת אחרת. אין להרוס את המחרוזת המקורית אלא להקצות זיכרון עבור תתי-המחרוזות המוחזרות.

דוגמה: בהרצת הפונקציה עם הפרמטרים הבאים:

```
str = "ThisIsANiceDayToGoToTheBeach"
separators = "Is", "A", "To", "The"
size = 4
```

נקבל כפלט את תתי-המחרוזות הבאות (ברשימה מקושרת):

"This", "Nice", "Day", "Go", "Beach"

שאלה מס' 4 (22 נקודות)

כתוב תכנית בשם `decode_it`. התכנית מקבלת כארגומנט שמות של קובצי טקסט, האחד ל-`input` והשני ל-`output`. קובץ ה-`input` מכיל מספרים שלמים בטווח שבין 0 ל- $2^{32}-1$, מופרדים על ידי רווחים לבנים (white spaces).

התכנית קוראת את הקובץ עד ל-`EOF`, מפענחת את הרצף, ומתרגמת אותו למחרוזת של תווי `ASCII`, אותה היא מדפיסה לקובץ ה-`output`. המחרוזת מורכבת מתווים, כאשר כל תו מתקבל ממספר מבין המספרים הרשומים בקובץ.

התרגום של מספר נתון i לכדי תו $c(i)$ מתבצע באופן הבא. נסמן ב- $one(i,k)$ את מספר המופעים של הספרה '1' בשמונת הביטים של ה-byte ה- k ייצוג הבינארי של i . ייצוג של מספר שלם מורכב מארבעה בתים כך ש- k נע בין 0-3. נסמן ב- $m(i)$ את מספר המקסימאלי של $one(i,k)$ עבור i נתון.


התו $c(i)$ מתקבל על ידי גישה למערך גלובלי `charcode` נגיש לתכנית, $c(i) = \text{charcode}[m(i)]$.

מחברת מס' 2
מתוך 2 מחברות

**לפני התחלת הבחינה מלא את כל הפרטים הבאים בכתב ברור
וקרא בעיון את ההוראות:**

1. על הנבחן להיבחן רק בחדר שבו הוא רשום.
2. עם הכניסה לחדר הבחינה יש להניח את החפצים בצד לדבות 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.
6. נבחן שנכנס לחדר הבחינה וקיבל את השאלון (טופס הבחינה) לידו ייחשב כמי שנבחן במועד זה. היה והחליט לא לכתוב את הבחינה, לא יהא רשאי לעזוב את חדר הבחינה, אלא כעבור חצי שעה ממועד תחילתה ולאחר שהחזיר את המחברת והשאלון. ציונו בבחינה יהיה "0".
7. קריאת השאלון מותרת רק לאחר קבלת רשות המשגיח.
8. יש לכתוב את התשובות בעט, בכתב יד ברור ונקי. נבחן הבוחר לכתוב טיוטה יעשה זאת בעמודו הימני של דפי מחברת הבחינה ויצוין בראש העמוד "טיוטה". אין לתלוש דפים מהמחברת.
9. מחברות הבחינה שקיבל הנבחן תהיינה בפיקוחו ובאחריותו במשך כל הבחינה. בעת יציאה מן החדר יופקדו המחברות והשאלון בידי המשגיח.
10. בתום הבחינה יחזיר הנבחן את המחברות והשאלון ויקבל מידי המשגיח את כרטיס הנבחן.
11. הנוהג בניגוד להוראות ולינוהל סדרי בחינות ודיווח ציונים" צפוי להפסקת בחינתו ואף להעמדה לדין משמעתי.
12. אין לכתוב מעבר לקו האדום משני צידי הדף.

לשימוש המורה הבוחן:

הציון 95
המחברת נבדקה ביום _____
חתימת המורה 

בהצלחה.

57074

תאריך הבחינה 16.2.05
שם הקורס מנהל 1
שם המורה פרופ' דניאל כהן-ארי
החוג/המנחה מנהל אפיצקה



void Merge(...) {

#include <stdio.h>
#include <stdlib.h>
#include <strings.h>

11

*curr-child
Tree-Node *tree-root;
List-Node *curr-par = *list-head;

int tmpind = 0;
*array-size = 0;

*array = (Merge-cell *) malloc(sizeof(Merge-cell));

(-1)

merge-cell * array -

while (curr-child

if (curr-child == NULL) return; /* no children in the class */

do { /* init per iteration */
*curr-par = *list-head;
(*array-size)++;

*array = (Merge-cell *) realloc(*array, (*array-size) * sizeof(Merge-cell));

tmpind = (*array-size) - 1; /* cuz i'm lazy :-> */

strcpy ((*array[tmpind]).sur-name, curr-child->sur-name);

strcpy ((*array[tmpind]).name, curr-child->name);

(*array[tmpind]).age = curr-child->age;

(*array[tmpind]).order-in-class = curr-child->order-in-class;

while (strcmp (curr-par->surname, curr-child->surname) != 0) {
curr-par = curr-par->next; /* find the parent */

(*array[tmpind]).avg-salary = curr-par->avg-salary;

(*array[tmpind]).is-divorced = curr-par->is-divorced;

} while (curr-child->next != NULL); /* cond after a step */

next
(-3)

}

הסרה: עזר במינה סדר לשם אר חיסוס הנורה לפני תחילת

הכתיבה למעורק, אבל מילון שהמכון בעל ואלו במחשב

לחוסם אר זה גרסיה רק יבאן.



מאגזין טמארטון



מאגזין טמארטון



מאגזין טמארטון



מאגזין טמארטון

המטרה: למצוא את הchild * /

```
curr-child = find-child(tempind+1);
/* handle by child num */
if (curr-child->order-in-class != tempind+1)
    curr-child = curr-child->left;
else curr-child = curr-child->right;
```

הנהגה: זכר הוראה המורה, להמשיך חזרה וקדימה - כל פעם באחד מהצדדים, הוראה שגור נ"י שכל מה שהכנס קודם לזה הנוסחאות - לאי שכל מה הקודם לזה הורה יותר קד, זה מה שזה הורה הנהגה.

TreeNode * find-child (TreeNode *head, int num) {

```
TreeNode *res;
TreeNode *curr = *head;
if (curr->order-in-class == num) return curr;
```

```
if (curr->right != NULL) find-child(curr->right, num);
else if (curr->left != NULL) find-child(curr->left, num);
```

```
else
    if (curr->right == NULL)
        if (find-child(curr->left, num) != NULL)
            res = find-child(curr->left, num);
        else if (curr->left == NULL) return find-child(curr->right, num);
        else return (find-child(curr->left, num) | find-child(curr->right, num));
        /* whichever address != NULL */
    if (curr->right == NULL && curr->left == NULL) return NULL;
    /* dead end */
}
```

(-2)
הנהגה: זכר הוראה המורה, להמשיך חזרה וקדימה - כל פעם באחד מהצדדים, הוראה שגור נ"י שכל מה שהכנס קודם לזה הנוסחאות - לאי שכל מה הקודם לזה הורה יותר קד, זה מה שזה הורה הנהגה.

(2)

```

int main(int argc, char **argv){
    FILE *ints_file, floats_file, target_file;

    int tmpint;
    float tmpfloat, epsilon, tmpdiff;
    char tmpchar;

    if (argc != 4) exit(1); /* bad input */
    ✓ epsilon = atof(argv[3]);
    ints_file = fopen(argv[0], "r+");
    floats_file = fopen(argv[1], "r+");
    target_file = fopen(argv[2], "w+");

    if (!ints_file || !floats_file) exit(1); /* file wouldn't open */
    if (!target_file) exit(1);
    while (fscanf(ints_file, "%d", &tmpchar) != EOF){
        fscanf(floats_file, "%f", &tmpfloat);
        tmpint = atoi(tmpchar);

        if ((tmpint - tmpfloat) >= epsilon || (tmpfloat - tmpint) >= epsilon) { /*
            tmpdiff = tmpint - tmpfloat;
            if (tmpdiff < 0) tmpdiff = 0 - tmpdiff;
            ✓ if (tmpdiff < epsilon){
                fprintf(target_file, "%f\n", tmpdiff); /* or "%d", depends on
                                                         the desired format */
            }
        }

        fclose(ints_file);
        ✓ fclose(floats_file);
        fclose(target_file);
        return 0;
    }
}
    
```

הקובץ
הוא "r+"
כלומר
ניתן לכתוב
וגם לקרוא

(-2)

זריקת
הקובץ
הוא "w+"
כלומר
ניתן לכתוב
וגם לקרוא

(1)

הקובץ
הוא "r+"
כלומר
ניתן לכתוב
וגם לקרוא

19
22

... הסיון קצ'ר

(2008)

```

Tree_Node* find-child (Tree_Node *root, char* surname){
    int cmpres;

return
    /* binary search, AT LAST */

    cmpres = strcmp(root->sur_name, surname);
    if (cmpres == 0) return root;
    if (cmpres < 0 && root->left != NULL)
        return return find-child(root->left, surname);
    if (cmpres > 0 && root->right != NULL)
        return find-child(root->right, surname);
    return NULL;
}
    
```

16

if (11)

```
tmpcell = (select_cell *) malloc(sizeof(select_cell));  
if (!tmpcell) exit(1);  
tmpcell->id = select_cell->id;  
tmpcell->parent = select_cell->parent;  
tmpcell->children = select_cell->children;  
tmpcell->value = select_cell->value;  
tmpcell->isLeaf = select_cell->isLeaf;  
tmpcell->isRoot = select_cell->isRoot;  
tmpcell->isPruned = select_cell->isPruned;  
tmpcell->isExpanded = select_cell->isExpanded;  
tmpcell->isVisited = select_cell->isVisited;  
tmpcell->isGoal = select_cell->isGoal;  
tmpcell->isFrontier = select_cell->isFrontier;  
tmpcell->isClosed = select_cell->isClosed;  
tmpcell->isOpen = select_cell->isOpen;  
tmpcell->isPruned = select_cell->isPruned;  
tmpcell->isExpanded = select_cell->isExpanded;  
tmpcell->isVisited = select_cell->isVisited;  
tmpcell->isGoal = select_cell->isGoal;  
tmpcell->isFrontier = select_cell->isFrontier;  
tmpcell->isClosed = select_cell->isClosed;  
tmpcell->isOpen = select_cell->isOpen;
```

WILLIAM H. HARRIS

王中书(王中书)

ה'תשנ"ב י"ב אלול

```
strcpy(tmpcell->sur_name, list_head->surname);
```

```
strcpy(tmpcell->name, tmpchild->name);
```

```
tmpchild->next = select(tree-root, list-head->next);
```

```
return temp tmpcell;
```

3

↳ else return ~~data~~ NULL; /* end of recursion */

}

(float *age;) ז'ר'ה קבוצה של מספרים

S/C : 1c . תכנת התיאטרון לא יום חג אלא גם שבת.

2. אם $\alpha \in \mathbb{R}$ אז $\alpha \in \mathbb{Q}$ או $\alpha \notin \mathbb{Q}$, אבל $\alpha \in \mathbb{Q}$ או $\alpha \notin \mathbb{Q}$

מ'11/11 2019 ימי'ה'ה' 2019 מ'11/11 2019

10/3/17 11/1/17 12/1/17 1/1/18 2/1/18 3/1/18 4/1/18 5/1/18 6/1/18 7/1/18 8/1/18 9/1/18 10/1/18 11/1/18 12/1/18 1/1/19 2/1/19 3/1/19 4/1/19 5/1/19 6/1/19 7/1/19 8/1/19 9/1/19 10/1/19 11/1/19 12/1/19 1/1/20 2/1/20 3/1/20 4/1/20 5/1/20 6/1/20 7/1/20 8/1/20 9/1/20 10/1/20 11/1/20 12/1/20 1/1/21 2/1/21 3/1/21 4/1/21 5/1/21 6/1/21 7/1/21 8/1/21 9/1/21 10/1/21 11/1/21 12/1/21 1/1/22 2/1/22 3/1/22 4/1/22 5/1/22 6/1/22 7/1/22 8/1/22 9/1/22 10/1/22 11/1/22 12/1/22 1/1/23 2/1/23 3/1/23 4/1/23 5/1/23 6/1/23 7/1/23 8/1/23 9/1/23 10/1/23 11/1/23 12/1/23 1/1/24 2/1/24 3/1/24 4/1/24 5/1/24 6/1/24 7/1/24 8/1/24 9/1/24 10/1/24 11/1/24 12/1/24 1/1/25 2/1/25 3/1/25 4/1/25 5/1/25 6/1/25 7/1/25 8/1/25 9/1/25 10/1/25 11/1/25 12/1/25 1/1/26 2/1/26 3/1/26 4/1/26 5/1/26 6/1/26 7/1/26 8/1/26 9/1/26 10/1/26 11/1/26 12/1/26 1/1/27 2/1/27 3/1/27 4/1/27 5/1/27 6/1/27 7/1/27 8/1/27 9/1/27 10/1/27 11/1/27 12/1/27 1/1/28 2/1/28 3/1/28 4/1/28 5/1/28 6/1/28 7/1/28 8/1/28 9/1/28 10/1/28 11/1/28 12/1/28 1/1/29 2/1/29 3/1/29 4/1/29 5/1/29 6/1/29 7/1/29 8/1/29 9/1/29 10/1/29 11/1/29 12/1/29 1/1/30 2/1/30 3/1/30 4/1/30 5/1/30 6/1/30 7/1/30 8/1/30 9/1/30 10/1/30 11/1/30 12/1/30 1/1/31 2/1/31 3/1/31 4/1/31 5/1/31 6/1/31 7/1/31 8/1/31 9/1/31 10/1/31 11/1/31 12/1/31 1/1/32 2/1/32 3/1/32 4/1/32 5/1/32 6/1/32 7/1/32 8/1/32 9/1/32 10/1/32 11/1/32 12/1/32 1/1/33 2/1/33 3/1/33 4/1/33 5/1/33 6/1/33 7/1/33 8/1/33 9/1/33 10/1/33 11/1/33 12/1/33 1/1/34 2/1/34 3/1/34 4/1/34 5/1/34 6/1/34 7/1/34 8/1/34 9/1/34 10/1/34 11/1/34 12/1/34 1/1/35 2/1/35 3/1/35 4/1/35 5/1/35 6/1/35 7/1/35 8/1/35 9/1/35 10/1/35 11/1/35 12/1/35 1/1/36 2/1/36 3/1/36 4/1/36 5/1/36 6/1/36 7/1/36 8/1/36 9/1/36 10/1/36 11/1/36 12/1/36 1/1/37 2/1/37 3/1/37 4/1/37 5/1/37 6/1/37 7/1/37 8/1/37 9/1/37 10/1/37 11/1/37 12/1/37 1/1/38 2/1/38 3/1/38 4/1/38 5/1/38 6/1/38 7/1/38 8/1/38 9/1/38 10/1/38 11/1/38 12/1/38 1/1/39 2/1/39 3/1/39 4/1/39 5/1/39 6/1/39 7/1/39 8/1/39 9/1/39 10/1/39 11/1/39 12/1/39 1/1/40 2/1/40 3/1/40 4/1/40 5/1/40 6/1/40 7/1/40 8/1/40 9/1/40 10/1/40 11/1/40 12/1/40 1/1/41 2/1/41 3/1/41 4/1/41 5/1/41 6/1/41 7/1/41 8/1/41 9/1/41 10/1/41 11/1/41 12/1/41 1/1/42 2/1/42 3/1/42 4/1/42 5/1/42 6/1/42 7/1/42 8/1/42 9/1/42 10/1/42 11/1/42 12/1/42 1/1/43 2/1/43 3/1/43 4/1/43 5/1/43 6/1/43 7/1/43 8/1/43 9/1/43 10/1/43 11/1/43 12/1/43 1/1/44 2/1/44 3/1/44 4/1/44 5/1/44 6/1/44 7/1/44 8/1/44 9/1/44 10/1/44 11/1/44 12/1/44 1/1/45 2/1/45 3/1/45 4/1/45 5/1/45 6/1/45 7/1/45 8/1/45 9/1/45 10/1/45 11/1/45 12/1/45 1/1/46 2/1/46 3/1/46 4/1/46 5/1/46 6/1/46 7/1/46 8/1/46 9/1/46 10/1/46 11/1/46 12/1/46 1/1/47 2/1/47 3/1/47 4/1/47 5/1/47 6/1/47 7/1/47 8/1/47 9/1/47 10/1/47 11/1/47 12/1/47 1/1/48 2/1/48 3/1/48 4/1/48 5/1/48 6/1/48 7/1/48 8/1/48 9/1/48 10/1/48 11/1/48 12/1/48 1/1/49 2/1/49 3/1/49 4/1/49 5/1/49 6/1/49 7/1/49 8/1/49 9/1/49 10/1/49 11/1/49 12/1/49 1/1/50 2/1/50 3/1/50 4/1/50 5/1/50 6/1/50 7/1/50 8/1/50 9/1/50 10/1/50 11/1/50 12/1/50 1/1/51 2/1/51 3/1/51 4/1/51 5/1/51 6/1/51 7/1/51 8/1/51 9/1/51 10/1/51 11/1/51 12/1/51 1/1/52 2/1/52 3/1/52 4/1/52 5/1/52 6/1/52 7/1/52 8/1/52 9/1/52 10/1/52 11/1/52 12/1/52 1/1/53 2/1/53 3/1/53 4/1/53 5/1/53 6/1/53 7/1/53 8/1/53 9/1/53 10/1/53 11/1/53 12/1/53 1/1/54 2/1/54 3/1/54 4/1/54 5/1/54 6/1/54 7/1/54 8/1/54 9/1/54 10/1/54 11/1/54 12/1/54 1/1/55 2/1/55 3/1/55 4/1/55 5/1/55 6/1/55 7/1/55 8/1/55 9/1/55 10/1/55 11/1/55 12/1/55 1/1/56 2/1/56 3/1/56 4/1/56 5/1/56 6/1/56 7/1/56 8/1/56 9/1/56 10/1/56 11/1/56 12/1/56 1/1/57 2/1/57 3/1/57 4/1/57 5/1/57 6/1/57 7/1/57 8/1/57 9/1/57 10/1/57 11/1/57 12/1/57 1/1/58 2/1/58 3/1/58 4/1/58 5/1/58 6/1/58 7/1/58 8/1/58 9/1/58 10/1/58 11/1/58 12/1/58 1/1/59 2/1/59 3/1/59 4/1/59 5/1/59 6/1/59 7/1/59 8/1/59 9/1/59 10/1/59 11/1/59 12/1/59 1/1/60 2/1/60 3/1/60 4/1/60 5/1/60 6/1/60 7/1/60 8/1/60 9/1/60 10/1/60 11/1/60 12/1/60 1/1/61 2/1/61 3/1/61 4/1/61 5/1/61 6/1/61 7/1/61 8/1/61 9/1/61 10/1/61 11/1/61 12/1/61 1/1/62 2/1/62 3/1/62 4/1/62 5/1/62 6/1/62 7/1/62 8/1/62 9/1/62 10/1/62 11/1/62 12/1/62 1/1/63 2/1/63 3/1/63 4/1/63 5/1/63 6/1/63 7/1/63 8/1/63 9/1/63 10/1/63 11/1/63 12/1/63 1/1/64 2/1/64 3/1/64 4/1/64 5/1/64 6/1/64 7/1/64 8/1/64 9/1/64 10

(13)

3

אני בלעז זמן יש לי סוף בראש.

האלגוריתם הבא:

א ← למצוא את המצבים הקטן ביותר שאינו null
והוא מוצא $\text{str}(\text{str}, \text{separator})$ (הוא המהיר)

בה יתן את המילה הבאה.

ב ← אם אין מצב, יתן את המילה הבאה.

הוא המילה הבאה קטנה מכל המילים האחרות.

למצוא את המילה הבאה (זה עם $\text{str}(\text{str})$)

ג ← מצבים את המילה הבאה word ← עם memory
זה next של המילה הבאה קטנה מכל המילים האחרות.

רקורסיבית wordparser

הוא מוצא את המילה הבאה str יתן את המילה הבאה.

הוא מוצא את המילה הבאה match (אם הוא match * (עם $\text{match} + \text{len}$)

אם עזר word אספר לראות שינוי לפי יסודות הקוד.

(אולי קצת יותר)

(3)

דאנקס, העלילה, וידע

Word * wordParser (...) {

~~int tmpword~~~~char * match~~char * match
Word * tmpword~~while (tmpword = strstr(str, separators[size-1]) != NULL)~~

if (strlen(str) == 0) return NULL; /* EOL */

match = strstr(str, separators[size-1])

if (match == NULL && size == 0) { /* stripped word */

tmpword = ~~malloc~~ (word *) malloc(sizeof(word));

if (!tmpword) exit(1);

~~tmpword = strcpy~~

strcpy(tmpword, str);

tmpword->next = wordParser(match + 1, separators, size);

}

if (match == NULL && size > 0) { /* finished with this tok */

wordParser(str, separators, size-1);

}

tmpword = (word *) malloc(sizeof(word));

if (!tmpword) exit(1);

wordParser(match, separators, size);

```

tmpind = size
while (match != null) {
    match = strstr(str,
    tmpind--;
    if tmpind == 0 . break;
}

```

4

```
void decode_it (char* inFile, char* outFile) {
```

```
FILE* inFile, outFile;
int tmpint;
```

20
22

for

```
inFile = fopen (inFile, "r");
```

```
outFile = fopen (outFile, "w");
```

```
if (!inFile || !outFile) exit (1); →
```

```
fscanf (inFile, "%d", &tmpint)
```

7
1208
777
777

1

```
while (fscanf (inFile, "%d", &tmpint) != EOF) {
```

✓

```
fprintf (outFile, "%c", charcode [max [tmpint]]);
```

```
}
```

```
fclose (inFile);
fclose (outFile);
```

```
}
```

return

מוגש ע"י תמרתון

מוגש ע"י תמרתון

מוגש ע"י תמרתון

מוגש ע"י תמרתון

הכנסת האחרונה של הקוד
 int max1 (int num) {
 int currmax = 0; tmp = 0;
 int mask = 1; i = 0; j = 0;

for (i = 0; i <= 3; i++) { /* running for 4 bytes */

 for (j = 0; j <= 8; j++) { /* going thru 1 byte */

 if (mask & num) tmp++;
 mask = mask << 1;
 }
 ✓

 if (tmp > currmax) currmax = tmp; /* comparing bytes' results */
}

return currmax;

}

... name recursion of (3)

Word* wordparser (...) {

char *match, *tmpmatch;

word *tmpword, j=0, i;
int tmpLen=0;

19/22

~~while (strstr(str, separators[j]) != NULL) j++;~~ while (strstr(str, separators[j]) != NULL) j++; /* ignore non-useful seps */
(0 & j < size)

k

/* dig down */

~~match = strstr(str, separators[j]);~~
match = strstr(str, separators[j]); tmpmatch = match;
for (i=j; i < size; i++) {

(3)

tmpmatch = strstr(str, separators[i]);

if (tmpmatch < match) { tmpmatch = match;
tmpLen = strlen(separators[i]);
}

/* now we have the smallest word possible */

tmpword = (word *) malloc(sizeof(word));

if (!tmpword) exit;

~~strcpy(tmpword, str);~~

2 (-2)

if (match == NULL) { /* end of recursion */

strcpy(tmpword, str);

return tmpword;

}

tmpword->next = NULL;

→ 37-1
1325
→ 5101

1 (-1)

~~strcpy(tmpword, str);~~ strcpy(tmpword, str, match - str);

tmpword->next = wordparser(match, separators, size);

return tmpword;

11111111
11111111
11111111
11111111
11111111

}

מחברת מס' _____
מתוך _____ מחברות

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

1. הנך מדרש לשמור על טוהר הבחינה ועל עבודה עצמית ולהישמע להוראות המשגיחים ולנוהלי האוניברסיטה. אין להעתיק, אין לדבר ואין להעביר חומר בין הנבחנים.

נבחן הנוהג בניכוד 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 מהמשגיח.

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

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

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

9. אין לתלוש 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 המשגיח את התעודת המזהה.

11. אין לכתוב מעבר לקו האדום משני צידי הדף.

תאריך הבחינה 06/02/05

שם הקורס 1 גנ"ב

שם המורה פרופ' דניאל בן-נחום

החונ/המגמה לפי' המסמך

לשימוש המורה הבוחן:

הציון 94
המחברת נבדקה ביום _____
חתימת המורה [Signature]

70203

בהצלחה.

```

int one(int i, int k)           // (4)
{
    int mask = 1; int idx; /*
    int one = 0; /*
    mask <= 4 * k;
    for (idx = 0; idx < 4; idx++)
    {
        if (i & mask != 0)
        {
            one++;
            mask <= 1;
        }
    }
    return one;
}

```

(4) mask <= 4 * k;
 (4) for (idx = 0; idx < 4; idx++)
 if (i & mask != 0)
 one++;
 mask <= 1;

mask <= 1;
 return one;

i & idx - ה-idx
 mask "0"
 (4 * k + idx)

```

int max_one(int i)           // m(i)
{
    int k, max = 0;
    int one;
    for (k = 0; k < 4; k++)
    {
        one = one(i, k);
        if (one > max)
        {
            max = one;
        }
    }
    return max;
}

```

(4) k = 0; k < 4; k++
 one = one(i, k);
 if (one > max)
 max = one;

return max;

```

int c(int i)
{
    return char code[ max_one(i) ];
}

```

char code[max_one(i)];



מוזכר מוגש ע"י



מוזכר מוגש ע"י



מוזכר מוגש ע"י



מוזכר מוגש ע"י

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

```
{ FILE *in, *out;
```

```
char i; char a; int c;
```

```
if (argc != 2) {  
    return -1;
```

קריאה נכונה
(8)

```
in = fopen (argv[0], "r")
```

```
if (in == NULL) return -1;
```

קריאה נכונה
פתיחה וקריאה
הקובץ

```
out = fopen (argv[1], "w")
```

```
if (out == NULL) return -1;
```

קריאה נכונה
פתיחה וקריאה
הקובץ

```
while ( (a = fscanf (in, "%d", &i)) != EOF)
```

קריאה נכונה
פתיחה וקריאה
הקובץ

```
{ c = c(i);
```

```
if (a = fputc (c, out) != EOF)
```

קריאה נכונה
פתיחה וקריאה
הקובץ

```
return -1;
```

```
}
```

```
if (a = fclose (in) != EOF)
```

```
return -1;
```

קריאה נכונה
פתיחה וקריאה
הקובץ

```
if (a = fclose (out) != EOF)
```

```
return -1;
```

```
return 0;
```

```
}
```

19/22

~~make-merge~~ וקריאה "Merge" של "List-Node" ו"Tree-Node" [16 G]

Merge-cell * make-merge (List-node * per, Tree-node * root)

{ char *sur; Tree-node *child; Merge-cell *m;

137 1st strcpy(sur, per->sur_name); ^{ה'סוף ה'סוף ה'סוף}

child = find-child(sur, Tree-node * root);

m = (merge-cell *) malloc (sizeof(merge-cell));

138 1st { strcpy(m->sur_name, sur);
strcpy(m->name, child->name);

(-3)

m->age = child->age;

m->order_in_class = child->order_in_class;

m->~~average~~ salary = per->^{avg}~~average~~ salary;

m->is-devised = per->is-devised;

מחלקת
התלמידים
המורה
מ-
המורה
המורה

return m;

{

Tree-node * find-child (char * sur, Tree-node * root)

{ Tree-node * p = root; int h;

139 ~~if~~ h = strcmp(sur, p->sur_name)

if (h == 0) return p;

if (h < 0)

return find-child(sur, p->left);

return find-child(sur, p->right);

ה'סוף ה'סוף
ה'סוף ה'סוף
ה'סוף ה'סוף
ה'סוף ה'סוף
ה'סוף ה'סוף

}

void merge (Tree-node *root, List-node *head, Merge cell
**a, int *size)

```
{ List-node *per = head;
  *size = 0; Merge cell *m;
```

while (per != NULL)

```
{ m = make-merge(per, Tree-node root);
```

```
*a[m->order-in-class] = m;
```

```
*size++;
```

```
per = per->next;
```

137
12

וידעו על יסודות
הקורס
ויקרו מהנה
למדת עמך
פע הורה והיטות
למדת א-

}

נשמח לשמוע!

רבים על יסודות הקורס, שמור כל הורה וקורס מהנה
"מחלק" שמורם על יסודות הקורס והמחלק מהנה (אם)
אם מחלק (הוספת הנהנה למקום הנהנה) המחלק
(אם מקום יסודות).

12

```

Select_Cell * make_Cell(List_Node *per, Tree_Node *root) (N 1)
{
    char *sur; Tree_Node *c;
    Select_Cell *s;

    strcpy(sur, per->sur_name);
    c = find-child(sur, root) // * * * * *
    s = (Select_Cell *) malloc(sizeof(Select_Cell));
    s->next = NULL;
    strcpy(s->sur_name, sur);
    strcpy(s->name, c->name);
    return c;
}

```

ייתכן שיש צורך ב-Select_Cell (N 1) * * * *

```

Select_Cell * Select(Tree_Node *root, List_Node *head)
{
    Select_Cell *first = NULL; // * * * * *
    Select_Cell *last = NULL; // * * * * *
    List_Node *per = head;

    while(per != NULL && per->avg_salary > 5000)
    {
        if(last == NULL) s = make_Cell(per, root);
        if(last == NULL)
        {
            head = s; last = s;
        }
        else { last->next = s; last = s; }
        per = per->next;
    }
    return first;
}

```

18

רק אם יש קישור ייחודי כל עוד
המספר < 5000 (ב) הקישור מוסיף
המספר יורד (המספר יורד) הורה יורד
אם המספר הוא ג'ס
ומספרים מוסיפים הורה יורד.

(א) הסינוי פ'רם למקד צה במקום הנצוקה פ'רם-מסוג
 מסימ short, פ'רמז'יק, מ'רמז'יק short (short +)
 נה עמבס לו מ'רמז'יק, מ'רמז'יק פ'רם י'רמז'יק
 מ'רמז'יק פ'רם מ'רמז'יק, מ'רמז'יק מ'רמז'יק. כ'רמז'יק נ'רמז'יק
 מ'רמז'יק (short) מ'רמז'יק (short) מ'רמז'יק
 מ'רמז'יק (short) מ'רמז'יק. כ'רמז'יק מ'רמז'יק
 מ'רמז'יק (short) מ'רמז'יק. כ'רמז'יק מ'רמז'יק
 מ'רמז'יק (short) מ'רמז'יק. כ'רמז'יק מ'רמז'יק

מ'רמז'יק (short) מ'רמז'יק. כ'רמז'יק מ'רמז'יק
 מ'רמז'יק (short) מ'רמז'יק. כ'רמז'יק מ'רמז'יק
 מ'רמז'יק (short) מ'רמז'יק. כ'רמז'יק מ'רמז'יק



10

2



מארגון תחרות



מארגון תחרות



מארגון תחרות



מארגון תחרות

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

```
{
    int a; char c;
    float b, e, res;
    File *in-int, *in-float, out;

```

~~if (fopen)~~



האם יש פתרון? (1)

האם יש פתרון?

```
if ( (in-int = fopen(argv[0], "rt")) || (in-float = fopen(argv[1], "rb")) )
    return -1;
```

```
if ( !(out = fopen(argv[2], "wb")) )
    return -1;
```

האם יש פתרון? (1)

```
e = argv[3];
```

* e - האם יש פתרון?

```
while ( fscanf(in-int, "%c", &c) != EOF )
{
    a = atoi(c);
```

```
fread(&b, sizeof(float), 1, in-float);
```

```
res = abs(float(b-a));
```

```
if ( res <= e )
```

```
fwrite(&res, sizeof(float), 1, out);
```

```
}
```

```
fclose(in-int);
```

```
fclose(in-float);
```

```
fclose(out);
```

```
return 0;
```

```
}
```

האם יש פתרון?

14/22

Word * make_word(char *S, int i, int j) (3)

```

{
    int size; int k;
    Word *new; char *C;
    size = j - i; // מילוי של המילה הקטנה
    new = (Word *) malloc(sizeof(char) * size);
    while(i <= j) C = (char *) malloc(sizeof(char) * size);
    for(k=0; k < size; k++) {
        C[k] = S[i+k];
    }
    new->str = C;
    new->next = NULL;
    return new;
}

```

* וי"ו "מילה" לא הנהיגו את ה"מילה" i-j

```

int find_sep(char *S, int i, char *separators)
{
    int k, j, l;
    for(k=0; k < size; k++)
    {
        sep = separators[k];
        T = strlen(sep);
        for(j=0; j < 1; j++)
        {
            for(i=0; i < 1 && S[i+j] == sep[j]; j++)
            {
                if(j == 1) return i;
            }
        }
    }
    return 0;
}

```

בוקר ואם יום מחולקת מכלי'ר החל לערוך ה"ס,

אם כן מחצית מן ה"ס

~~מחצית מן ה"ס~~



Word * wordParser (char *s, char **separators, int size)

```
{ word * head = NULL, * last = NULL, * w;
  int start, end = 0; int i
```

```
  start = 0;
```

```
  do {
```

```
    while (!find_sep(s, end, separators, size))
```

```
      end++
```

```
    w = make_word(s, start, end);
```

```
    if (last == NULL)
```

```
      { head = w; last = w; }
```

```
    else { last->next = w;
```

```
          last = w; }
```

```
    start = end + 1;
```

```
    end = start;
```

```
  } while (end < strlen(s))
```

```
  return head;
```

```
}
```

במקרה של המחרוזת, כלומר במסלול המחרוזת

המחרוזת, בתוספת גם המחרוזת, המחרוזת והמחרוזת

ומחרוזת המחרוזת המחרוזת.

& end < strlen(s)

(-1)

אין סימן > מניחים

קידום עם המחרוזת המחרוזת

21/22