

תכנות מערכות בשפת C ושפת סף

הוראות לנבחן

א. משך הבחינה: ארבע שעות.

ב. מבנה השאלון ומפתח ההערכה: בשאלון זה שני נושאים: **תכנות מערכות בשפת C ושפת סף** ובהם שמונה שאלות. עליך לענות על **שש** שאלות, על-פי ההנחיות בכל פרק. בשני הנושאים בסך-הכול – 100 נקודות.

ג. חומר עזר מותר לשימוש: כל חומר עזר כתוב בכתב-יד או מודפס על נייר.

ד. לנוחותך, לשאלון זה מצורף מילון מונחים בשפות עברית, אנגלית, רוסית וערבית. תוכל להיעזר בו בעת הצורך.

בשאלון זה 42 עמודים ו-2 עמודי נספחים.

ההנחיות בשאלון זה מנוסחות בלשון זכר, אך מכוונות הן לנבחנות והן לנבחנים.

בהצלחה!

המשך מעבר לדף ◀

השאלות

נושא א': תכנות מערכות בשפת C (50 נקודות)

פרק ראשון (40 נקודות)

ענה על שתי השאלות 1-2.

שאלה 1 – שאלת חובה (25 נקודות)

חברת השמה במרכז הארץ מתמקדת בקישור בין חברות הייטק שמעוניינות לקלוט מהנדסים ובין מועמדים מתאימים. החברה הקימה מערכת ממוחשבת העוקבת אחר העסקה של עובדים, לרבות הפסקת עבודתם. מספר החברות המטופלות על-ידי חברת ההשמה הוא NUM_OF_COMPANIES והן ממוספרות מ-1 ועד NUM_OF_COMPANIES.

מהנדס המעוניין להגיש מועמדותו לעבודה מקבל טופס מועמדות מחברת ההשמה. המהנדס ממלא בטופס את פרטיו האישיים ואת דרישת השכר שלו שגדולה מאפס, ונקלט במערכת הממוחשבת במעמד של "בלתי מועסק". במערכת הממוחשבת קיים מאגר הכולל פרטי מהנדסים בלתי מועסקים ופרטי מהנדסים שמועסקים באחת מ-NUM_OF_COMPANIES החברות.

הנחות יסוד:

1. NUM_OF_COMPANIES הוא מספר קבוע ואינו משתנה מרגע האתחול.
2. מהנדס מועסק בחברת הייטק אחת בלבד.
3. כל חברת הייטק קובעת סף מקסימלי לשכר מהנדס בחברתה.
4. מהנדס מתקבל לעבודה בתנאי שדרישת השכר שלו אינה עולה על השכר המקסימלי שקבעה החברה, והשכר שיקבל בפועל בעבור עבודתו יהיה בהתאם לדרישת השכר שציין בטופס המועמדות.

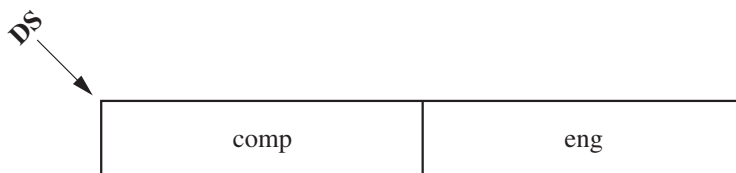
לפניך תיאור של **מבנה הנתונים** אשר יתמוך במימוש הפעולות הנדרשות מן המערכת הממוחשבת על-ידי חברת ההשמה (הדרישות יפורטו בהמשך).

נחזיק מבנה (רשומה) שנכנה אותו בשם "**המבנה הראשי**" הכולל את השדות האלה:

שדה 1 – comp – מערך של חברות הייטק בגודל `NUM_OF_COMPANIES`. כל תא במערך זה מייצג חברה הייטק מסוימת. התא ה- `i` במערך הוא מבנה המכיל את פרטי חברת ההייטק שמספרה `i+1`.

שדה 2 – eng – מערך דינמי של כל המהנדסים הרשומים במערכת הממוחשבת (מועסקים או בלתי מועסקים). כל תא במערך זה הוא מבנה המכיל את פרטי המהנדס הרשום בחברת ההשמה.

איור א' שלהלן מציג תיאור סכמתי של "**המבנה הראשי**":



איור א' לשאלה 1

"**המבנה הראשי**" מאגד את כל מבני הנתונים שבהם נאחסן את כל נתוני המערכת הממוחשבת.

להלן הגדרת קבוע המציין את מספר חברות ההייטק:

```
#define NUM_OF_COMPANIES 5
```

ולהלן הגדרת **המבנה הראשי** בשפת C:

```
typedef struct headType // טיפוס המבנה הראשי
{
    companyPtr comp; // מערך חברות הייטק
    engineerPtr eng; // מערך דינמי – פרטי מהנדסים
}header, *headPtr;
```

עֵתָה נִפְרָט אֶת מִבְנֵה הַנְּתוּנִים בַּעֲבוּר שִׁדָּה 1 שֶׁל "הַמִּבְנֵה הָרִאשִׁי".

לְהֵלֵךְ מִבְנֵה שֶׁל תֵּא בַּמַּעְרָךְ **comp** בַּשִּׁפְת C :

```
typedef struct companyType // טיפוס חברת הייטק
{
    int companyID; // מספר החברה
    int numOfEmp; // מספר המהנדסים המועסקים בחברה זו
    long salaryThd; // שכר מקסימלי שחברה מוכנה לשלם למהנדס
    employeePtr empList; // מצביע לראש רשימה מקושרת חד-כיוונית המייצגת
    // מהנדסים המועסקים בחברה זו
} companyRec, *companyPtr;
```

לְהֵלֵךְ מִבְנֵה שֶׁל צוּמַת בְּרִשִׁימַת הַמַּהֲנָדָסִים שֶׁמַּעֲסִיקָה חֲבֵרַת הַיִּיטָק מִסוּיִמַת:

```
typedef struct employeeType // טיפוס צומת ברשימה
{
    long engID; // מספר זהות של מהנדס
    long salary; // דרישת שכר מהנדס
    struct employeeType *next; // שדה קישור
} employeeRec, *employeePtr;
```

הערות:

1. בַּמַּעֲרֶכֶת הַמִּמּוֹחֶשֶׁבֶת קִיִּימַת רִשִׁימַת מַהֲנָדָסִים נִפְרָדַת לְכָל חֲבֵרַת הַיִּיטָק.
2. הַרִשִׁימוֹת הָאֵלֶּה מִמּוִינּוֹת בְּאוֹפֵן בִּלְתִּי תִּלּוֹי זֶה בְּזֶוּ, בִּסְדֵּר יוֹרֵד, עַל פִּי מִפְתָּח שֶׁל שֹׁכֵר מַהֲנָדָס בַּחֲבֵרָה זֶוּ.

עתה נפרט את מבנה הנתונים בעבור **שדה 2 של "המבנה הראשי"**.

להלן מבנה של **תא במערך דינמי eng** בשפת C :

```
typedef struct engineerType //(מועסק או בלתי מועסק)
{
    long engineerID; // מספר זהות של מהנדס
    char lastName[20]; // שם משפחה
    char firstName[15]; // שם פרטי
    char tel[10]; // מספר טלפון
    long reqSalary; // דרישת שכר
    int compID; // שדה זה יכיל את מספר החברה שבה הוא מועסק
                // שדה זה יכיל את הערך 0
}engineerRec , *engineerPtr;
```

הערה:

המערך eng ממזין בסדר עולה, על פי מפתח שהוא מספר הזהות של המהנדס.

להלן הגדרות של משתנים גלובאליים:

```
header head;
headPtr DS = &head;
DS->eng=NULL;
DS->comp=NULL;
int numOfEngineers = 0; // מספר המהנדסים הרשומים במערכת הממוחשבת של חברת ההשמה
```

להלן הגדרות התקפות לכל הסעיפים שיבואו בהמשך:

```
typedef enum  
{FAILURE, SUCCESS, INVALID_INPUT, ALLOCATION_ERROR, DUPLICATE_ID,  
ID_NOT_EXIST, ID_ALREADY_HIRED, SALARY_NOT_FIT, EMPTY_ARRAY}  
statusType;  
  
typedef enum {FALSE, TRUE} boolean;
```

נתונה ספריית פונקציות המכילה בין היתר את הפונקציה הזאת:

<p>פונקציה זו מקבלת את מערך המהנדסים – rp ואת key – שהוא מספר הזהות של מהנדס מסוים. אם key קיים במערך rp, אזי הפונקציה מחזירה את האינדקס של התא במערך המכיל את פרטי המהנדס הזה; אחרת – הפונקציה מחזירה את הערך -1.</p>	<pre>int findEngineer(engineerPtr rp, long key)</pre>
--	---

הנח שפונקציה זו כתובה וניתן להשתמש בה בכל הסעיפים הבאים בלי לכתוב אותה מחדש.

ענה על הסעיפים הבאים:

א. לפניך פונקציה שכותרתה:

```
statusType registerEngineer(long engineerID, char lastName[],  
char firstName[], char tel[], long reqSalary)
```

פונקציה זו מקבלת את פרטיו של מהנדס חדש שהגיש מועמדות לחברת ההשמה ואת דרישת השכר שלו.

הפונקציה מבצעת את הצעדים האלה:

- מגדילה את מערך המהנדסים ומוסיפה לסוף מערך זה את המהנדס החדש, בתנאי שהוא אינו קיים במערכת הממוחשבת.
 - מעדכנת את מספר המהנדסים במערכת הממוחשבת.
- נוסף על כך, הפונקציה **ממיינת** את מערך המהנדסים באמצעות הפונקציה `qsort` אשר משתמשת בפונקציה `longcmp`.

פונקציה זו מחזירה ערך מטיפוס `statusType`, כמפורט בטבלה שלהלן:

ALLOCATION_ERROR	אם קיימת בעיה בהקצאת זיכרון.
INVALID_INPUT	אם דרישת השכר של המהנדס היא ערך שלילי.
DUPLICATE_ID	אם המהנדס שמספר הזהות שלו <code>engineered</code> קיים במערכת הממוחשבת.
SUCCESS	אם המהנדס שמספר הזהות שלו <code>engineered</code> אינו קיים במערכת הממוחשבת, דרישת השכר שלו גדולה או שווה ל-0 ולא הייתה קיימת בעיה בהקצאת זיכרון.

בפונקציה חסרים **שישה** ביטויים, המסומנים במספרים בין סוגריים עגולים. רשום במחברת הבחינה את מספרי הביטויים החסרים (1) – (6), בסדר עולה, וכתוב ליד כל מספר את הביטוי החסר שהוא מייצג.

```
statusType registerEngineer(long engineerID, char lastName[], char
                             firstName[], char tel[], long reqSalary)
{
    statusType status = SUCCESS;
    int pindx;
    if (reqSalary < 0) return INVALID_INPUT;
    pindx = _____ (1) _____;
    if (pindx == -1) //New engineer
    {
        DS->eng = _____ (2) _____;
        if (_____ (3) _____) return ALLOCATION_ERROR;
        DS->eng[numOfEngineers].engineerID = engineerID;
        strcpy(DS->eng[numOfEngineers].lastName, lastName);
        strcpy(DS->eng[numOfEngineers].firstName, firstName);
        strcpy(DS->eng[numOfEngineers].tel, tel);
        DS->eng[numOfEngineers].reqSalary = reqSalary;
        _____ (4) _____ = 0;
        numOfEngineers = numOfEngineers + 1;
        _____ (5) _____;
    }
    else
        status = _____ (6) _____;
    return status;
}
```



```
}  
  
int longcmp(constvoid *p,constvoid *q)  
{  
    engineerPtr t1,t2;  
    long n1,n2;  
    t1 = p;  
    t2=q;  
    n1 = t1->engineerID;  
    n2 = t2->engineerID;  
    return n1-n2;  
}
```

ג. לפיך פונקציה שכותרתה:

```
statusType adjustCompany(long engineerID,int cmpID,long reqSalary)
```

פונקציה זו מקבלת את הפרמטרים:

engineerID – מספר הזהות של מהנדס

cmpID – מספר חברת ההייטק

reqSalary – דרישת השכר של מהנדס זה

פונקציה זו מוסיפה לרשימה מקושרת חד-כיוונית ממוינת של חברת ההייטק שמספרה cmpID צומת חדש בעבור מהנדס שמספר הזהות שלו הוא engineerID וכן מעדכנת את מספר העובדים המועסקים בחברה זו.

לאחר הוספת הצומת, הרשימה תישאר ממוינת בסדר יורד על פי מפתח שכר המהנדסים.

הנח כי מהנדס זה אינו מועסק בחברה זו ודרישת השכר שלו מתאימה לחברה.

פונקציה זו מחזירה ערך מטיפוס `statusType`, כמפורט בטבלה שלהלן:

ALLOCATION_ERROR	אם קיימת בעיה בהקצאת זיכרון.
SUCCESS	אם המהנדס שמספר הזהות שלו <code>engineerID</code> נוסף בהצלחה לרשימת המהנדסים בחברה זו.

בפונקציה חסרים **חמישה** ביטויים, המסומנים במספרים בין סוגריים עגולים. רשום במחברת הבחינה את מספרי הביטויים החסרים (1) – (5), בסדר עולה, וכתוב ליד כל מספר את הביטוי החסר שהוא מייצג.

`statusType adjustCompany(long engineerID,int cmpID,long reqSalary)`

```
{
    statusType status = SUCCESS;
    employeePtr t,p,q;

    t = DS->comp[cmpID-1].empList;
    p = malloc(sizeof(employeeRec));
    if (p==NULL)
    {
        status = ALLOCATION_ERROR;
        return status;
    }
    p->engID = engineerID;
    p->salary = reqSalary;
    p->next = NULL;
    if (t==NULL)
    {
        _____(1)_____ = p;
    }
}
```

```
else
    if (t->salary < reqSalary)
    {
        p->next = _____ (1) _____;
        _____ (1) _____ = p;
    }
else
{
    q = t;
    t = t->next;
    while(t)
    {
        if(t->salary < reqSalary)
        {
            _____ (2) _____ = t;
            _____ (3) _____ = p;
            break;
        }
        q = t;
        t = t->next;
    }
    if(t==NULL) _____ (4) _____;
}
_____ (5) _____;
return status;
}
```

ג. לפניך פונקציה שכותרתה:

```
statusType hire(int company, long engineerID)
```

פונקציה זו מקבלת כפרמטר את company – מספרה של חברת הייטק, ואת engineerID – מספר הזהות של מהנדס.

הפונקציה מצרפת את המהנדס שמספר הזהות שלו הוא engineerID לחברת ההייטק שמספרה company, בתנאי שהמהנדס רשום במערכת הממוחשבת של חברת ההשמה, מוגדר כבלתי מועסק ועומד בתנאי השכר של החברה.

פונקציה זו מחזירה ערך מטיפוס statusType, כמפורט בטבלה שלהלן:

ALLOCATION_ERROR	אם קיימת בעיה בהקצאת זיכרון.
ID_NOT_EXIST	אם המהנדס אינו רשום במערכת הממוחשבת.
ID_ALREADY_HIRED	אם המהנדס כבר מועסק.
SUCCESS	אם המהנדס שמספר הזהות שלו engineerID נוסף בהצלחה לרשימת המהנדסים בחברה שמספרה company.

בפונקציה חסרים **ארבעה** ביטויים, המסומנים במספרים בין סוגריים עגולים. רשום במחברת הבחינה את מספרי הביטויים החסרים (1) – (4), בסדר עולה, וכתוב ליד כל מספר את הביטוי החסר שהוא מייצג.

```
statusType hire(int company, long engineerID)
```

```
{
    statusType status = SUCCESS;
    int indx;
    engineerPtr p;
    indx = _____ (1) _____;
    if(indx == -1) return ID_NOT_EXIST;
    p = DS->eng + indx;
    if(_____ (2) _____) return ID_ALREADY_HIRED;
    _____ (3) _____;
    status = _____ (4) _____;
    return status;
}
```

4. לפניך פונקציה שכותרתה:

```
statusType hireBySalary(int company)
```

פונקציה זו מקבלת כפרמטר את company – המספר של חברת ההייטק שמעוניינת להעסיק מהנדס שעונה על הדרישות האלה:

- רשום בחברת ההשמה כבלתי מועסק.
- דרישת השכר שלו אינה עולה על סף השכר המקסימלי שקבעה החברה.

הפונקציה מצרפת לרשימת המהנדסים המועסקים בחברה שמספרה company את המהנדס שדרישת השכר שלו היא הגבוהה ביותר מבין כל המהנדסים שעונים לדרישות שלעיל.

שים לב: ייתכן שלא ימצא מהנדס כזה.

הנחה:

במערכת הממוחשבת של חברת ההשמה קיימים מהנדסים בלתי מועסקים.

פונקציה זו מחזירה ערך מטיפוס statusType, כמפורט בטבלה שלהלן:

אם קיימת בעיה בהקצאת זיכרון.	ALLOCATION_ERROR
אם מספר חברת ההייטק אינו תקין.	INVALID_INPUT
אם המהנדס אינו רשום במערכת הממוחשבת.	ID_NOT_EXIST
אם המהנדס מועסק.	ID_ALREADY_HIRED
אם במערכת הממוחשבת לא נמצא מהנדס בלתי מועסק שעומד בדרישות השכר של החברה company.	SALARY_NOT_FIT
אם המהנדס נוסף בהצלחה לרשימת המהנדסים בחברה זו.	SUCCESS

בפונקציה חסרים **ארבעה** ביטויים, המסומנים במספרים בין סוגריים עגולים. רשום במחברת הבחינה את מספרי הביטויים החסרים (1) – (4), בסדר עולה, וכתוב ליד כל מספר את הביטוי החסר שהוא מייצג.

```
statusType hireBySalary(int company)
{
    statusType status = SUCCESS;
    long ms, req, ID;
    long req1, ID1;
    int i;
    req = 0;
    if(____(1)____ || company > NUM_OF_COMPANIES)
        return INVALID_INPUT;
    ms = DS->comp[company - 1].salaryThd;
    ID = -1;
    for(i=0; i < numOfEngineers; i++)
    {

        req1 = DS->eng[i].reqSalary;
        ID1 = DS->eng[i].engineerID;
        if(____(2)____ || req1 > ms) continue;
        if ((req1 > req))
        {
            req= req1; ID = ID1;
        }
    }
    if (____(3)____) status =SALARY_NOT_FIT;
    else
        status =____(4)____;

    return status ;
}
```

ה. לפניך פונקציה שכותרתה:

```
statusType fireEmployee(int companyID,long engineerID)
```

פונקציה זו מקבלת כפרמטר את companyID – מספרה של חברת הייטק כלשהי, ואת engineerID – מספר הזהות של מהנדס כלשהו.

הפונקציה מטפלת בתהליך הפסקת עבודתו של המהנדס שמספרו engineerID בחברת ההייטק שמספרה companyID. מעתה, מהנדס זה יהיה רשום במערכת הממוחשבת של חברת ההשמה כמהנדס בלתי מועסק.

נוסף על כך, הפונקציה **מעדכנת** את מספר המועסקים בחברת ההייטק שמספרה companyID.

פונקציה זו מחזירה ערך מטיפוס statusType, כמפורט בטבלה שלהלן:

אם בחברת ההייטק שמספרה companyID לא מועסק מהנדס שמספרו engineerID, או שהחברה לא מעסיקה כלל מהנדסים, או שהחברה לא קיימת במערכת הממוחשבת.	FAILURE
אם המהנדס אינו רשום במערכת הממוחשבת.	ID_NOT_EXIST
אם הופסקה עבודתו של מהנדס שמספר הזהות שלו הוא engineerID בחברת ההייטק שמספרה companyID, אז הוא יופיע במערכת הממוחשבת כמהנדס בלתי מועסק.	SUCCESS

בפונקציה חסרים **ארבעה** ביטויים, המסומנים במספרים בין סוגריים עגולים. רשום במחברת הבחינה את מספרי הביטויים החסרים (1) – (4), בסדר עולה, וכתוב ליד כל מספר את הביטוי החסר שהוא מייצג.

```
statusType fireEmployee(int companyID,long engineerID)
{
    statusType status=FAILURE;
    int indx;
    employeePtr p,q;
    indx = findEngineer(DS->eng,engineerID);
    if (indx == -1) return ID_NOT_EXIST;
    if (DS->eng[indx].compID != companyID) return FAILURE;
    if (companyID < 1 || companyID > NUM_OF_COMPANIES) return FAILURE;
    _____ (1) _____ = 0;
    p = DS->comp[companyID -1].empList;
    q = p;
    p = p->next;
    if(q->engID==engineerID)
    {
        _____ (2) _____ = p;
        free(q);
        status = SUCCESS;
    }
    else
        while(p)
        {
            if(_____ (3) _____)
```



```

{
    q->next = p->next;
    free(p);
    status = SUCCESS;
    break;
}

q = p;
p = p->next;
}

if (status == SUCCESS) _____ (4) _____;
return status;
}

```

1. לפניך פונקציה שכותרתה:

`statusType cancelRegistration(long engineerID)`

פונקציה זו מקבלת כפרמטר את `engineerID` – מספר הזהות של מהנדס בלתי מועסק הרשום במערכת הממוחשבת.

הפונקציה מוחקת מהנדס זה מן המערכת הממוחשבת של חברת ההשמה.

הפונקציה מחזירה ערך מטיפוס `statusType`, כמפורט בטבלה שלהלן:

אם לאחר מחיקת מהנדס זה מן המערכת הממוחשבת לא קיימים בה מהנדסים נוספים.	EMPTY_ARRAY
אם המהנדס שמספר הזהות שלו הוא <code>engineer</code> אינו קיים במערכת הממוחשבת של חברת ההשמה, או שהמהנדס מועסק באחת מחברות ההייטק.	ID_NOT_EXIST
אם המהנדס שמספר הזהות שלו הוא <code>engineer</code> נמחק בהצלחה מן המערכת הממוחשבת של חברת ההשמה, ועדיין נותרו מהנדסים במערכת.	SUCCESS

שים לב: יש לדאוג שהמערך `eng` יישאר ממוין לאחר ביצוע פעולה זו.

בפונקציה חסרים שני ביטויים, המסומנים במספרים בין סוגריים עגולים. רשום במחברת הבחינה את מספרי הביטויים החסרים (1) – (2), בסדר עולה, וכתוב ליד כל מספר את הביטוי החסר שהוא מייצג.

```
statusType cancelRegistration(long engineerID)
{

    statusType status = SUCCESS;
    int indx;
    engineerPtr p,q;

    indx = findEngineer(DS->eng,engineerID);
    p = DS->eng +indx;
    if (indx != -1 && p->compID == 0)
    {
        q = DS->eng + numOfEngineers - 1;
        p->engineerID = q->engineerID;
        strcpy(p->lastName,q->lastName);
        strcpy(p->firstName,q->firstName);
        strcpy(p->tel,q->tel);
        p->compID = q->compID;
        p->reqSalary = q->reqSalary;
        numOfEngineers--;
        DS->eng = _____ (1) _____;
        if(DS->eng ==NULL) return EMPTY_ARRAY;
        _____ (2) _____;
    }
    else status = ID_NOT_EXIST;

    return status;
}
```

ז. במערכת הממוחשבת קיימים שני קבצים בינאריים:

1. קובץ בשם "HTCompanies.dat" המכיל את פרטי חברות ההייטק.
2. קובץ בשם "Engineers.dat" המכיל את פרטי המהנדסים המועסקים או הבלתי מועסקים.

להלן הגדרות של מבנים:

מבנה (רשומה) לצורך קריאת נתוני חברת הייטק מקובץ "HTCompanies.dat"

```
struct company
{
    int companyID; // מספר חברה
    int numOfEmp; // מספר המהנדסים המועסקים בחברה זו
    long salaryThd; // שכר מקסימלי שחברה זו מוכנה לשלם למהנדס
} companyin;
```

מבנה (רשומה) לצורך קריאת נתוני מהנדס (מועסק או בלתי מועסק) מקובץ "Engineers.dat"

```
struct engineer
{
    long engineerID; // מספר הזהות של מהנדס
    char lastName[20]; // שם משפחה
    char firstName[15]; // שם פרטי
    char tel[10]; // מספר טלפון
    long reqSalary; // דרישת שכר
    int compID; // בעבור מהנדס מועסק - שדה זה יכיל את מספר החברה שבה הוא מועסק,
    // בעבור מהנדס מובטל - שדה זה יכיל את הערך 0
} engineerin;
```

לפניך פונקציה שכותרתה:

```
statusType init_engineer(void)
```

הפונקציה קוראת פרטי מהנדסים (מועסקים ובלתי מועסקים) מן הקובץ "Engineers.dat".
בעבור כל מהנדס היא משכנת את פרטיו במערך המהנדסים - eng, ובעבור מהנדס מועסק היא
גם מצרפת אותו לרשימת המהנדסים של החברה שבה הוא מועסק.

הנח כי פרטי החברות נקראו מקובץ "HTCompanies.dat" ושוכנו במערך החברות - comp.

פונקציה זו מחזירה ערך מטיפוס statusType, כמפורט בטבלה שלהלן:

ALLOCATION_ERROR	אם קיימת בעיה בהקצאת זיכרון.
INVALID_INPUT	אם דרישת השכר של המהנדס היא ערך שלילי.
DUPLICATE_ID	אם המהנדס שמספר הזהות שלו engineerID קיים כבר במערכת הממוחשבת.
SUCCESS	אם המהנדס שמספר הזהות שלו engineerID לא קיים כבר במערכת הממוחשבת, דרישת השכר שלו גדולה או שווה ל-0, ולא הייתה קיימת בעיה בהקצאת זיכרון.

בפונקציה חסרים **שלושה** ביטויים, המסומנים במספרים בין סוגריים עגולים. רשום במחברת
הבחינה את מספרי הביטויים החסרים (1) - (3), בסדר עולה, וכתוב ליד כל מספר את הביטוי
החסר שהוא מייצג.

```
statusType init_engineer(void)
```

```
{
```

```
FILE *fp;
```

```
statusType status = SUCCESS;
```

```
long engineerID;
```

```
char lastName[20];
```

```
char firstName[15];
```

```
char tel[10];
```

```
long compID;
```

```
int reqSalary;

int i=0;

struct engineer *engbuff = &engineerin;

/*****

* Load engineers Table *

*****/

if ((fp = fopen("Engineers.dat", "rb")) == NULL)
{
    puts("\n Error opening file.\n");
    exit(1);
}

_____ (1) _____;

while (!feof(fp))
{
    engineerID = engineerin.engineerID;
    strcpy(lastName, engineerin.lastName);
    strcpy(firstName, engineerin.firstName);
    strcpy(tel, engineerin.tel);
    reqSalary = engineerin.reqSalary;
    compID = engineerin.compID;
    status = _____ (2) _____;
    if ((compID) && (status == SUCCESS))
        status = _____ (3) _____;
    _____ (1) _____;
}

fclose(fp);

return status;

}
```

שאלה 2 - שאלת חובה (15 נקודות).
בשאלה זו ארבעה סעיפים בלתי תלויים זה בזה. ענה על כולם.

א. נתונה התכנית הבאה בשפת C :

```
#include<stdio.h>

typedef const char cchr;

void func(int res[],int a[],int b[],int N);
void wrt(cchr *s,int a[],int size);

int main(void)
{
    int a[] = {0,1,9,3,0,5}; // המספר שמשוכן ב־a הוא 503910
    int b[] = {0,0,2,0,0,0}; // המספר שמשוכן ב־b הוא 000200
    int res[(sizeof(a)+sizeof(b))/sizeof(a[0])] = {0};
    int N=(sizeof(a))/sizeof(a[0]);
    func(res,a,b,N);
    wrt("a:",a,sizeof(a)/sizeof(a[0]));
    wrt("b:",b,sizeof(b)/sizeof(b[0]));
    wrt("res:",res,(sizeof(a)+sizeof(b))/sizeof(a[0]));
    return 0;
}

void func(int res[],int a[],int b[],int N)
{
    int carry = 0;
    inti,j;

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

```
{
    res[i+N-1] += carry;
    if (res[i+N-1] < 10)
        carry = 0;
    else
    {
        carry = res[i+N-1]/10;
        res[i+N-1] %= 10;
    }
}
for(j=0;j<N;j++)
{
    res[i+j] += b[i]*a[j] + carry;
    if(res[i+j] < 10)    carry = 0;
    else
    {
        carry = res[i+j]/10;
        res[i+j] %=10;
    }
}
}
```

```
void wrt(cchr *s,int a[],int size)
{
    int i=size-1;
    printf("%s",s);
    for(;i>=0;--i)
        printf("%d",a[i]);
    putchar('\n');
}
```

- I. רשום במחברתך את הפלט המדויק של התכנית הנתונה.
- II. נתונים שני מספרים ארוכים כאשר הספרות של המספר הראשון משוכנות בסדר הפוך במערך a , והספרות של המספר השני משוכנות בסדר הפוך במערך b .
- להלן חמישה היגדים שאחד מהם מתאר את הפעולה שמבצעת התכנית על שני המספרים הנתונים. רשום במחברתך את מספרו של ההיגד הנכון.
- **היגד 1:** התכנית מכפילה את שני המספרים האלה ומדפיסה את מכפלתם.
 - **היגד 2:** התכנית מחברת את שני המספרים האלה ומדפיסה את תוצאת החיבור.
 - **היגד 3:** התכנית מחלקת את a ב- b ומדפיסה את מנת החילוק.
 - **היגד 4:** התכנית מחלקת את a ב- b ומדפיסה את שארית החילוק.
 - **היגד 5:** התכנית מחשבת את $a - b$ ומדפיסה את תוצאת ההפרש.

ב. לפניך תכנית בשפת C. רשום במחברתך את הפלט המדויק של התכנית הנתונה.

```
#include <stdlib.h>

#include <stdio.h>

int main()

{
    int **mat = NULL;

    int i = 0;

    mat = (int**) malloc(3 * sizeof(int*));

    mat[0] = (int*) malloc(10 * sizeof(int));

    for (i=0; i<10; i++) mat[0][i] = (i+1)*6;

    mat[1] = mat[0]+2;

    mat[2] = mat[1]+2;

    printf("A: %d\n", mat[0][0]);

    printf("B: %d\n", mat[1]-mat[0]);

    printf("C: %d\n", mat[1][3]);

    printf("D: %d\n", mat[2][2]);
```



```
printf("E: %d\n", *((mat+2)+3));

free(mat[0]);

free(mat);

return 0;

}
```

ג. נתונה ההגדרה הבאה:

```
char *p[2][3]={ "Mamamia", "Zahava", "good", "luck", "for", "me" };
```

לפניך ארבע פקודות הדפסה שאינן תלויות זו בזו. בעבור כל פקודת הדפסה רשום במחברת הבחינה את הפלט המדויק.

1. `printf("%s\n" , *p[0]+2);`
2. `printf("%s\n" , *(p[0]+2));`
3. `printf("%c\n" , *(p[1][1]+2));`
4. `printf("%s\n" , *p[1]);`

ד. לפניך התכנית הבאה הכוללת את הפונקציות `func` ו-`main`. בפונקציה `main` מוגדר משתנה `str` שהוא מערך מצביעים למחרוזות. הנח כי:

- `str[0]` מצביע למחרוזת שאיננה ריקה.
- התא האחרון של `str` מכיל את הזקף `NULL`.

הפונקציה `func` צריכה לבדוק אילו מהמחרוזות מוכלות במלואן וברצף בתוך המחרוזת הראשונה המוצבעת על ידי `str[0]`. הפונקציה מחזירה מחרוזת **חדשה**, שהיא שרשור של כל המחרוזות האלה.

שים לב: גם המחרוזת המוצבעת על ידי `str[0]` מוכלת בעצמה.

לדוגמה:

עבור `char* str[]={"1234","4","134","23","321",NULL};`

תתקבל המחרוזת החדשה הבאה (קרא משמאל לימין): "1234423".

הפונקציה `main` מדפיסה את המחרוזת החדשה, המוחזרת על ידי הפונקציה `func`, תוך שימוש בפונקציה `print`.

בתכנית חסרים **תשעה** ביטויים, המסומנים במספרים בין סוגריים עגולים. רשום במחברת הבחינה את מספרי הביטויים החסרים (1) – (9), בסדר עולה, וכתוב ליד כל מספר את הביטוי החסר שהוא מייצג.

```
#include <stdio.h>

#include <stdlib.h>

#include <string.h>

char* func(_____(1)_____ A)
{
    int i=1;

    char * p = NULL;

    p=_____(2)_____ ; // משכפל קודם את הראשון

    if(!p){printf("not enough memory!");exit(1);}

    while(_____(3)_____)
    {
        if(_____(4)_____)
        {
            p=(char*)realloc(p,_____(5)_____);

            if(!p){printf("not enough memory!");exit(2);}
        }
    }
}
```

מוסיף את המחרוזת למערך התוצאה// ; _____ (6) _____

}

_____ (7) _____ ;

}

return p;

}

void print(const char* str)

{

_____ (8) _____;

}

void main()

{

char* str[]={"1234","4","134","23","321",NULL};

char* p=func(_____ (9) _____);

print(p);

free(p);

}

פרק שני (10 נקודות)

ענה על אחת מבין השאלות 3-4 (לכל שאלה – 10 נקודות).

שאלה 3

לפניך תכנית בשפת C :

```
#include <stdio.h>

void main()
{
    int i,res=0;
    unsigned int a,x;
    unsigned char b;
    a=0x4a73;
    b=0x9;
    x= 0xf;
    i=0;
    do
    {
        if  (!(((a&x)>>i)^b))  res++;
        x = x<<1;
        i++;
    } while ((x>>28) ^ 0x0e);
    printf ("the number is : %d\n",res);
    return 0;
}
```

הנח כי טיפוס int מיוצג באמצעות ארבעה בתים (32 סיביות).

א. מה תדפיס התכנית עבור ערכי a, b הנתונים? רשום במחברתך את הפלט המדויק.

ב. מה תדפיס התכנית עבור ערכי a, b הבאים (כתחליף להשמה המתבצעת בתכנית):
 $a = 0xffff$, $b = 0x7$? רשום במחברתך את הפלט המדויק.

ג. מה תדפיס התכנית עבור ערכי a, b הבאים (כתחליף להשמה המתבצעת בתכנית):
 $a = 0xa557$, $b = 0xa$? רשום במחברתך את הפלט המדויק.

ד. מה תדפיס התכנית עבור ערכי a, b הבאים (כתחליף להשמה המתבצעת בתכנית):
 $a = 0x7c7e$, $b = 0xe$? רשום במחברתך את הפלט המדויק.

ה. כתוב במחברתך את המספר שליד התשובה הנכונה.
התכנית מונה ומדפיסה את:

1. מספר הספרות הזוגיות כסדרת ביטים ב a .
2. מספר הפעמים שמופיע $0x7$ כסדרת ביטים ב a .
3. מספר הספרות האי־זוגיות כסדרת ביטים ב a .
4. מספר הפעמים שמופיעה b כסדרת ביטים ב a .
5. מחצית מספר הפעמים שמופיעה b כסדרת ביטים ב a .

שאלה 4

לפניך הגדרה של צומת ברשימה מקושרת חד-כיוונית בשפת C :

```
typedef struct node_type
{
    int info;
    struct node_type *next;
}*node_ptr;
```

נתונה רשימה בגודל n כאשר ערכו של n אינו ידוע, אבל ידוע כי ערכו של n מתחלק ב-3 ושברשימה ישנם לפחות שלושה איברים.

לפניך פונקציה הנעזרת בשלושה מצביעים בלבד וללא מונים, ומדפיסה את השליש האמצעי של הרשימה. ראשית, הפונקציה סורקת את הרשימה **פעם אחת בלבד** ומגלה את השליש האמצעי של הרשימה, ולאחר מכן, היא מדפיסה את איברי הרשימה השייכים לשליש האמצעי שלה.

```
void printmidlist(node_ptr *p)
{
    node_ptr *p1,*p2,*p3;
    p1=p;
    p2=p1->next;
    p3=p2->next;

    while ( _____ (1) _____ != NULL)
    {
        p1=p1->next;
        p2= _____ (2) _____;
        p3= _____ (3) _____;

    }
```

```

p1=p1->next;

p2=_____ (4) _____;

while (_____ (5) _____)

{

    int x=_____ (6) _____;

    printf("%d",x);

    _____ (7) _____;

}

}

```

בפונקציה זו חסרים **שבעה** ביטויים המסומנים במספרים בין סוגריים עגולים (1) – (7) . בכל אחד מן הסעיפים שלהלן, בחר את הביטוי החסר מבין ארבע האפשרויות הנתונות. רשום במחברתך את אות הסעיף וציין לידה את מספר התשובה הנכונה.

א. הביטוי החסר (1) הוא:

1. p3->next
2. p2
3. p3
4. p1->next

ב. הביטוי החסר (2) הוא:

1. p2->info
2. p2->next->next
3. p2->next->next->next
4. p1->next->next

ג. הביטוי החסר (3) הוא:

1. $p2 \rightarrow info$
2. $p3 \rightarrow next \rightarrow next$
3. $p3 \rightarrow next \rightarrow next \rightarrow next$
4. $p2 \rightarrow next \rightarrow next \rightarrow next$

ד. הביטוי החסר (4) הוא:

1. $p1 \rightarrow next$
2. $p2 \rightarrow next \rightarrow next$
3. $p2 \rightarrow next$
4. $p1$

ה. הביטוי החסר (5) הוא:

1. $p3! = p1$
2. $p2! = p1$
3. $p2! = NULL$
4. $p1! = NULL$

ו. הביטוי החסר (6) הוא:

1. $p3 \rightarrow info$
2. $p1 \rightarrow info$
3. $p2 \rightarrow info$
4. $p2 \rightarrow next \rightarrow info$

ז. הביטוי החסר (7) הוא:

1. $p1=p1->next$

2. $p3=p3->next$

3. $p2=p2->next$

4. $p1->next->next$

נושא ב': שפת סף (50 נקודות)

פרק שלישי (20 נקודות)

ענה על שאלה 5 – שאלת חובה.

שאלה 5

לפניך תכנית בשפת אסמבלי הכוללת שגרה **רקורסיבית** בשם E_SUM אשר מקבלת באמצעות מחסנית את NUM, שהוא מספר עשרוני שלם חיובי הגדול מאפס.

הנחת יסוד: ערכו של המספר שמשוכן במשתנה NUM הוא 2559 לכל היותר.

```
SSEG SEGMENT STACK 'STACK'
```

```
DB 100H DUP(?)
```

```
SSEG ENDS
```

```
CODE SEGMENT
```

```
ASSUME CS:CODE
```

```
NUM DW 1353
```

```
START: PUSH NUM
```

```
CALL E_SUM
```

```
MOV AH,4CH
```

```
INT 21H
```

```
E_SUM: PUSH BP
```

```
MOV BP,SP
```

```
MOV AX,[BP+4]
```

```
MOV BL,10
```

```
DIV BL
```

```
OR AL,AL
```

```
JZ STOP_E_SUM
```

```

MOV [BP+4], AH

MOV BYTE PTR [BP+5], 0

XOR AH, AH

PUSH AX

CALL E_SUM ; (1)

STOP_E_SUM: TEST BYTE PTR [BP+4], 1

JNZ ODD

INC AL

ODD: POP BP

RET 2

CODE ENDS

END START

```

- א. מה יהיה תוכנו של האוגר AL, בבסיס עשרוני, לאחר ביצוע התכנית הנתונה?
- ב. אם נחליף בתכנית הנתונה את השורה NUM DW 1353 בשורה NUM DW 1087 – מה יהיה תוכנו של האוגר AL, בבסיס עשרוני, לאחר ביצוע התכנית?
- ג. אם נחליף בתכנית הנתונה את השורה NUM DW 1353 בשורה NUM DW -64183 – האם תוכנו של האוגר AL, בבסיס עשרוני, יהיה שונה מהערך שהוחזר באוגר AL בסעיף א'? ענה "כן" או "לא".
- ד. להלן ארבעה היגדים שאחד מהם מתאר את הפעולה שמבצעת התכנית הנתונה. רשום במחברתך את מספרו של ההיגד הנכון.
 - היגד 1:** השגרה הרקורסיבית מחזירה את הערך 0 אם המספר NUM הוא בבסיס בינארי, ומחזירה את הערך 1 אם המספר NUM הוא בבסיס עשרוני.
 - היגד 2:** השגרה הרקורסיבית מחזירה מבין הספרות הזוגיות ב־NUM את ערך הספרה הגדולה ביותר.
 - היגד 3:** השגרה הרקורסיבית מחזירה את הערך 0 אם ערכה של הספרה הגדולה ביותר ב־NUM הוא אי־זוגי; אחרת הפונקציה מחזירה את הערך 1.
 - היגד 4:** השגרה הרקורסיבית מתייחסת למספר שב־NUM כאל מספר עשרוני ומחזירה את מספר הספרות הזוגיות בו.

- ה. האם השגרה הרקורסיבית הנתונה מתייחסת למספר העשרוני שב-`NUM` כאל מספר לא מסומן? ענה "כן" או "לא".
- ו. אם נחליף בתכנית הנתונה את השורה `DIV BL` בשורה `IDIV BL` – האם שינוי זה ישפיע על הערך המוחזר באוגר `AL`? ענה "כן" או "לא".
- ז. אם נחליף בתכנית הנתונה את השורה `CALL E_SUM` המסומנת ב-(1), בשורה `JMP E_SUM` – האם שינוי זה ישפיע על הערך המוחזר באוגר `AL`? ענה "כן" או "לא".
- ח. אם נחליף בתכנית הנתונה את השורה `TEST BYTE PTR[BP+4],1` בשורה `AND BYTE PTR[BP+4],1` – האם ביצועי התכנית ישתנו בהכרח? ענה "כן" או "לא".
- ט. אם נחליף בתכנית הנתונה את השורה `RET 2` בשתי השורות הבאות:

`RET`

`RET`

האם ביצועי התכנית ישתנו בהכרח? ענה "כן" או "לא".

פרק רביעי (30 נקודות)

ענה על שתיים מבין השאלות 6–8 (לכל שאלה – 15 נקודות).

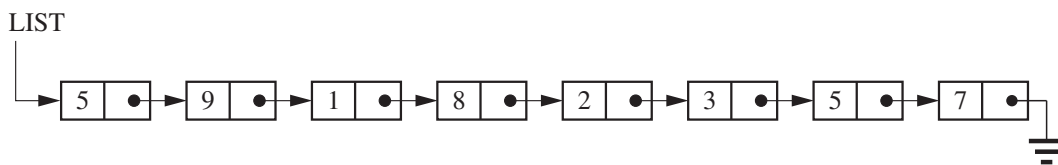
שאלה 6

נתונה רשימה מקושרת חד-כיוונית **לא ריקה** הבנויה מצמתים. כל צומת ברשימה מכיל את שני השדות האלה:

info – שדה מידע (אינפורמציה), שגודלו 8 ביטים, המכיל מספר שלם הגדול מאפס (מספר חסר סימן).

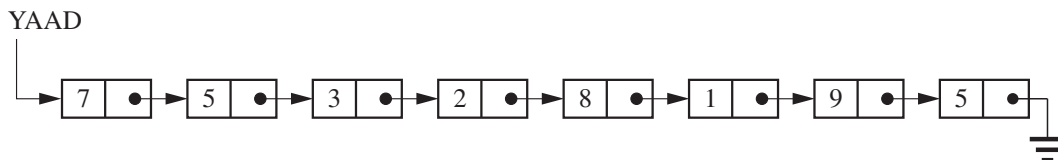
next – המצביע אל הצומת הבא ברשימה, שגודלו מילה אחת (16 ביטים). המצביע `next` בצומת האחרון הוא 0.

נוסף על כך, נתון כי המשתנה `LIST` הוא מצביע לצומת הראשון ברשימה, כלומר `LIST` מכיל את הכתובת של הצומת הראשון ברשימה (ראה איור א').



איור א' לשאלה 6

לפניך קטע תכנית בשפת אסמבלי אשר סורק את הרשימה הנתונה ובונה רשימה חדשה אשר תכיל את האיברים שנמצאים ברשימה הנתונה אך בסדר הפוך.
המשתנה YAAD יצביע לצומת הראשון ברשימה החדשה, כלומר יכיל את הכתובת של הצומת הראשון ברשימה (ראה איור ב').
בעבור הרשימה שבאיור א' קטע התכנית יבנה את הרשימה הבאה:



איור ב' לשאלה 6

לביצוע משימה זו קטע התכנית משתמש במחסנית.
נוסף על כך, הנח שקיימת שגרה בשם INSEND אשר מקבלת באמצעות מחסנית את:

- YAAD
- NUM – מספר לא מסומן

השגרה יוצרת צומת חדש, מציבה בשדה ה־info של צומת זה את המספר NUM ומוסיפה את הצומת הזה לסוף הרשימה החדשה אשר לראשה מצביע YAAD.

שים לב: השגרה INSEND מסתיימת בפקודה RET 4.

בקטע התכנית הנתון חסרים **שישה** ביטויים המסומנים במספרים בין סוגריים עגולים.

רשום במחברתך את מספרי הביטויים החסרים (1) – (6) **בלבד**, בסדר עולה, וכתוב ליד כל מספר את הביטוי החסר שהוא מייצג.

```
_____ (1) _____  
XOR      CX, CX  
  
AGAIN: MOV    AL, [BX]  
  
_____ (2) _____  
INC      CX  
  
CMP      _____ (3) _____, 0  
JE       CONT  
  
_____ (4) _____  
JMP      AGAIN  
  
CONT:  
  
_____ (5) _____  
CALL     INSEND  
  
GO:      _____ (6) _____
```

שאלה 7

לפניך תכנית בשפת אסמבלי שמוגדר בה משתנה NUM אשר גודלו מילה אחת (16 סיביות). הערך הנמצא בכל בית במילה זו גדול מאפס.

```
DATA SEGMENT  
  
    NUM    DW 0809H  
  
    M      DB ?  
  
    S      DB ?  
  
DATA ENDS  
  
SSEG SEGMENT STACK 'STACK'
```

```
DB 100H DUP(?)

SSEG ENDS

CODE SEGMENT

    ASSUME CS:CODE,DS:DATA

START:  MOV AX,DATA
        MOV DS,AX
        PUSH NUM
        CALL P_DIV
        POP WORD PTR M

SOF:    MOV AH,4CH
        INT 21H

P_DIV:  PUSH BP
        MOV BP,SP
        MOV AX,[BP+4]
        MOV BYTE PTR[BP+4],0
        CMP AH,AL
        JAE CON
        XCHG AL,AH

CON:    SUB AH,AL
        JC FINISH
        INC BYTE PTR[BP+4]
        MOV [BP+5],AH
        JMP CON

FINISH: POP BP
        RET

CODE ENDS

END START
```

א. להלן חמש טענות אשר אחת מהן מתארת את הפעולה שמבצעת התכנית הנתונה. רשום במחברתך את מספרה של הטענה הנכונה.

- 1: התכנית סופרת את מספר הסיביות המכובות (שערך 0) בשני בתיו של NUM.
- 2: התכנית מחשבת את השארית לאחר חילוקו של הבית הגבוה של NUM בבית הנמוך של NUM.
- 3: התכנית מתייחסת למספר המשוכן ב- NUM כאל שני מספרים, נניח a ו- b , וכל אחד מהם בגודל בית. אם $a \geq b$, אז התכנית מחזירה ב- M את המנה של a/b , וב- S את השארית של a/b . אחרת, התכנית מחזירה ב- M את המנה של b/a וב- S את השארית של b/a .
- 4: התכנית מתייחסת למספר המשוכן ב- NUM כאל שני מספרים, נניח a ו- b , וכל אחד מהם בגודל בית. אם $a \geq b$, אז התכנית מחזירה ב- S את המנה a/b , וב- M את השארית של a/b . אחרת, התכנית מחזירה ב- M את המנה b/a וב- S את השארית של b/a .
- 5: התכנית מתייחסת למספר המשוכן ב- NUM כאל שני מספרים, נניח a ו- b , וכל אחד מהם בגודל בית. אם $a \geq b$, אז התכנית מחזירה ב- M את המנה a/b וב- S את $a*b$. אחרת, התכנית מחזירה ב- M את המנה b/a וב- S את $b*a$. נוסף על כך, אם $a*b$ עולה על 255, אז S יקבל את הערך 1; ולא – את הערך 0.

עבור כל אחד מן הסעיפים ב'–ה', רשום במחברתך את הערכים המשוכנים במשתנה M ובמשתנה S , בבסיס הקסאדצימלי (בסיס 16), בהגיע התכנית לתווית SOF – בהתאם לאמור בכל סעיף.

- ב. כאשר מריצים את התכנית כפי שהיא.
- ג. כאשר מחליפים בתכנית הנתונה את השורה NUM DW 0809H בשורה NUM DW 0908H.
- ד. כאשר מחליפים בתכנית הנתונה את השורה NUM DW 0809H בשורה NUM DW 0408H.
- ה. כאשר מחליפים בתכנית הנתונה את השורה NUM DW 0809H בשורה NUM DW 0904H.

שאלה 8

הגדרה: פלינדרום הוא מילה, מספר, משפט או כל רצף סמלים שאם נקרא אותו מימין לשמאל או משמאל לימין נקבל את אותו הרצף.

דוגמאות למילים שהן פלינדרום: "זוז", "הסוסה", "היפהפיה" ו-"טרימוירט"

דוגמאות למספרים עשרוניים שהם פלינדרום: 1457541, 999999, 123321.

דוגמאות למספרים בינאריים שהם פלינדרום: 10100101, 11111111.

נתונה מטריצה בשם MAT של סיביות בגודל 8×8 , כלומר, כל איבר במטריצה זו הוא ביט (סיבית אחת). לפניך תכנית אשר מחזירה במשתנה RESULT את הערך 1 כאשר האיברים של האלכסון הראשי מהווים פלינדרום; ולא – התכנית מחזירה במשתנה RESULT את הערך 0.

בתכנית הנתונה חסרים **תשעה** ביטויים המסומנים במספרים בין סוגריים עגולים. רשום במחברתך את מספרי הביטויים החסרים (1) – (9) **בלבד**, בסדר עולה, וכתוב ליד כל מספר את הביטוי החסר שהוא מייצג.

```
DATA SEGMENT
    MAT DB 128,64,32,16,8,4,2,1
    RESULT DB 1
DATA ENDS
SSEG SEGMENT STACK 'STACK'
    DB 100H DUP(?)
SSEG ENDS
CODE SEGMENT
    ASSUME CS:CODE,DS:DATA
START: MOV AX,DATA
        MOV DS,AX
        LEA BX,MAT
        LEA DI,MAT+7
        MOV AL,128 ; AL==10000000B
```

```
MOV AH, 1 ; AH==00000001B
AG: MOV DL, [BX]
    _____ (1) _____
    MOV DH, [DI]
    _____ (2) _____
    MOV CL, DL
    OR DL, DH
    _____ (3) _____
    MOV DL, CL
    _____ (4) _____ DL, DL
    JZ BAD
    _____ (4) _____ DH, DH
    JZ BAD
OK: _____ (5) _____
    _____ (6) _____
    INC BX
    _____ (7) _____
    CMP BX, DI
    _____ (8) _____
    JMP AG
BAD: _____ (9) _____
FINISH: MOV AH, 4CH
        INT 21H
CODE ENDS
END START
```

בהצלחה!

נספח: מילון מונחים (2 עמודים)

לשאלון 714001, אביב תשע"ה

תרגום המונח			המונח
אנגלית	רוסית	ערבית	
			תכנות מערכות בשפת C :
initialization	Инициализация	إبتداء	אתחול
composition	Включение в себя, содержание в себе	ضمّ	הכלה
memory allocation	выделение памяти	تخصيص ذاكرة	הקצאת זיכרון
flag	Обозначение конца, конечный элемент	عَلَم	זקיף
one-direction	однонаправленный	ذات اتجاه واحد	חד-כיווני
type	Тип	نوع	טיפוס
data structure	структура данных	بُنية البيانات	מבנה נתונים
dynamic array	Динамический массив	مصفوفة غير ثابتة	מערך דינמי
pointer	Указатель	مُؤشّر	מצביע
global variable	Глобальная переменная	متغيّر عام	משתנה גלובאלי
bits series	Последовательность битов	سلسلة bit	סדרת ביטים
parameter	Параметр	متغيّر (بارامتر)	פרמטר
node	Узел, вершина	مَفْرَق	צומת
fixed	Константа	ثابت	קבוע
binary file	Двоичный файл	مِلَف ثنائي	קובץ בינארי
linked list	связный список	قائمة مرتبطة	רשימה מקושרת
field	поле	حَقْل	שדה
concatenation	Конкатенация	تَرَابُط	שרשור
cell	Ячейка	خَلِيّة	תא

תרגום המונח			המונח
אנגלית	רוסית	ערבית	
			שפת סף:
register	Регистр	مخزن (ريغستر)	אוגר
main diagonal	Главная диагональ	مائل رئيسي	אלכסון ראשי
hexadecimal base	Шестнадцатеричная система счисления, основание 16	قاعدة الست عشرية	בסיס הקסאדצימלי
stack	Стек	باغة	מחסנית
decimal base number	десятичное число	عدد عشري	מספר בבסיס עשרוני
marked number	обозначенное число, число со знаком	عدد مُعَلَّم	מספר מסומן
bits	биты	البتات	סיביות
palindrome	Палиндром	بوليندروم	פלינדרום
routine	Функция, рутина	رتابة	שגרה
recursive routine	Рекурсивная функция	إجراء تراجعي	שגרה רקורסיבית
string	Метка	نصّ	תווית