

סטודנטים יקרים

לפניכם ספר תרגילים בקורס ארגון ותכנות המחשב (שפת אסמבלי). הספר הוא חלק מפרויקט חדשני וראשון מסוגו בארץ במקצוע זה, המועבר ברשת האינטרנט On-line. הקורס באתר כולל פתרונות מלאים לספר התרגילים. הקורס נכתב ע"י גברת שלי קול.

הפתרונות מוגשים בסרטוני וידאו המלווים בהסבר קולי, כך שאתם רואים את התהליכים בצורה מובנית, שיטתית ופשוטה, ממש כפי שנעשה בשיעור פרטי.

אז אם אתם עסוקים מידי בעבודה, סובלים מלקויות למידה, רוצים להצטיין או פשוט אוהבים ללמוד בשקט בבית, אנחנו מזמינים אתכם לחוויית לימודים יוצאת דופן וחדשה לחלוטין, היכנסו עכשיו לאתר



אנו מאחלים לכם הצלחה מלאה בבחינות

צוות האתר GooL

גול, בשביל התירגול!

לפתרון מלא בסרטון וידאו היכנסו ל- www.Gool.co.il

© GOOL כל הזכויות שמורות לאתר

תרגיל 1 - תכנית ראשונה

כתוב תכנית אסמבלי המדפיסה "Hello World!" למסך.

תרגיל 2 - פעולות השבוניות ופקודות קפיצה

כתוב תכנית השואלת את המשתמש מה הציון שלו במבוא למדעי המחשב. לאחר מכן, התוכנית קולטת מספר בין 0 ל 100.
אם הציון הוא 100 התכנית תדפיס "You are the best" אם הציון מעל 90 התכנית תדפיס "You are good" לכל מקרה אחר, "Try to get more ...".

תרגיל 3 - התייחסות לזיכרון

השלם את הטבלה הבאה, אם הפקודה חוקית (תחת 8086), רשום את משמעותה.

הפקודה	חוקית Yes/No	משמעות
MOV [AX],DL		
MOV [BX],CL		
MOV AX, [SI]		
MOV [DI-23],CL		
MOV [DI+SI],CH		
MOV [BX],[DI]		
MOV [SI],DH		
MOV DS:[DI],AX		
MOV CL,1234		

לפתרון מלא בסרטון וידאו היכנסו ל- www.Gool.co.il

MOV DH,300		
MOV CH,011H		
MOV CH,DX		
MOV AX,BL		
MOV AL,AH		
MOV DL,256		
MOV DH,255		
MOV BX, OFFSET x		
MOV [BL],AL		
MOV SI, 2 MOV DI, OFFSET str[SI]		

לפתרון מלא בסרטון וידאו היכנסו ל- www.Gool.co.il

© GOOL כל הזכויות שמורות לאתר

תרגיל 4 - מספרים מסומנים

יש לקלוט מהמשתמש מספר בינארי בצורה של ארבעה ביטים. המספר יכול להיות שלילי (בשיטת המשלים ל-2) או חיובי. אם המספר הנקלט חיובי, יש להדפיס את השלילי המתאים לו בשיטת המשלים ל-2. אחרת, יש להדפיס הודעה מתאימה.

דוגמא לפלט התוכנית:

```
THIS PROGRAM CONVERTS TO NEGATIVE.
```

```
Please enter binary number:
```

```
0101
```

```
Your Number is Positive. Its Negative is: 1011
```

דוגמא נוספת:

```
THIS PROGRAM CONVERTS TO NEGATIVE.
```

```
Please enter binary number:
```

```
1111
```

```
Your Number is Negative!!!
```

תרגיל 5 – לולאות

כתוב תוכנית המגדירה ב-DATA את המערך הבא :

אינדקס	7	6	5	4	3	2	1	0
ערך	121	-67	31	-76	11777	-5867	1125	-2000

על התכנית למצוא ולהדפיס את הממוצע של המספרים במערך. ניתן להניח כי הממוצע הוא מספר שלם.

לפתרון מלא בסרטון וידאו היכנסו ל- www.Gool.co.il

תרגיל 6 – פרוצדורות

סעיף א:

כתוב פרוצדורה באסמבלי המחשבת מקסימום במערך של מספרים שלמים.

יש לכתוב את הפרוצדורה כך ש :

1. הכתובת לאיבר הראשון במערך נתונה ב DI.

2. מספר האיברים במערך (גודל המערך) נתון ב-CX.

סעיף ב:

כתוב פרוצדורה באסמבלי המדפיסה מספר שלם.

יש לכתוב את הפרוצדורה כך שהפרמטר (המספר השלם) נתון בשטח זיכרון גלובלי בשם num.

סעיף ג:

הוסף MAIN לתוכנית כך שידפיס את המקסימום במערך הבא :

2443	45	6	-45	-3456	5678	56	-89	-4	321	5688	43
------	----	---	-----	-------	------	----	-----	----	-----	------	----

גודל המערך : 12.

לפתרון מלא בסרטון וידאו היכנסו ל- www.Gool.co.il

תרגיל 7 - אסמבלי + שפת C

כתבו פרוצדורה באסמבלי הנקראת משפת C ומקבלת מערך של מספרים שלמים (מסוג int), את גודל המערך וכן מערך נוסף ופויינטר לגודל המערך הנוסף. הפרוצדורה מחשבת את סכום המספרים במערך ובודקת האם קיימים מספרים במערך הגדולים מהסכום. אם כן, על הפרוצדורה להחזיר 1 ולהעתיק את המספרים הגדולים מהסכום למערך הנוסף ולאתחל את גודלו. אחרת, על הפרוצדורה להחזיר 0. הגדרת הפרוצדורה היא:

```
int SumBigger(int arr[], int size, int bigArr[], int * bigSize);
```

קוד C הקורא לפרוצדורה:

```
#include <stdio.h>

#define N 10

extern int SumBigger (int arr[], int size, int bigArr[], int * bigSize);

void main()
{
    int arr[N], bigArr[N], bigSize, i;
    printf("\nPlease enter %d numbers\n", N);
    for(i=0;i<N;i++) scanf("%d",&arr[i]);
    if(SumBigger (arr, N, bigArr, &bigSize)) {
        printf("\nNumbers that larger than the sum:\n");
        for (i=0; i<bigSize; i++) printf("%d  ", bigArr[i]);
    } else printf("\nThere are NO numbers larger than the sum\n");
    return;
}
```

לפתרון מלא בסרטון וידאו היכנסו ל- www.Gool.co.il

תרגיל 8 - מודלים של זיכרון, פויינטר לפונקציה
סעיף א':

ממש פרוצדורה בשפת אסמלי שתקרא מתוך תכנית C המבצעת חיבור של מספרים במערך A עם המספרים במערך B והתוצאה תהיה במערך C.
ז"א, הפונקציה תבצע $C[i]=A[i]+B[i]$.
בנוסף, על הפונקציה להחזיר את הסכום הכולל $(\sum_{i=0}^{size} A[i] + B[i])$.
יש לממש את הפרוצדורה במודל LARGE.

הגדרת הפרוצדורה היא: `int add_nums(int A[],int B[], int C[], int size);`

קוד C הקורא לפרוצדורה הוא:

```
#include <stdio.h>

#define N 10

extern int add_nums(int A[],int B[], int C[], int size);

void main()
{
    int A[N], B[N], C[N], i, size=N;
    printf("\nEnter %d Numbers\n", N);
    for(i=0; i<N; i++) scanf("%d",&A[i]);
    printf("\nAgain, Enter %d Numbers\n", N);
    for(i=0; i<N; i++) scanf("%d",&B[i]);
    printf("\n The result is %d\n", add_nums(A, B, C, size));
    printf("\n The result Array is:\n");
    for(i=0; i<N; i++) printf("%d\t", C[i]);
}
```

לפתרון מלא בסרטון וידאו היכנסו ל- www.Gool.co.il

סעיף ב' :

שנה את הגדרת הפרוצדורה add_nums, כך שתקבל פויינטר לפונקציית איתחול מערך.

פונקציית האיתחול: void init arr(int arr[], int size);

הגדרת add_nums החדשה:

```
int add_nums(int A[],int B[], int C[], int size, void (*initf)(int[], int));
```

קוד C הקורא לפרוצדורה הוא:

```
#include <stdio.h>

#define N 10

extern int add_nums(int A[],int B[], int C[], int size,
                    void (*initf)(int[], int));
void initarr(int arr[], int size);

void main() {

    int A[N], B[N], C[N], i, size=N;
    printf("\n The result is %d\n", add_nums(A, B, C, size, initarr));
    printf("\n The result Array is:\n");
    for(i=0; i<N; i++) printf("%d\t", C[i]);

}

void initarr(int arr[], int size) {

    int i;
    printf("\nEnter %d Numbers\n", size);
    for(i=0; i<size; i++) scanf("%d",&arr[i]);

}
```

לפתרון מלא בסרטון וידאו היכנסו ל- www.Gool.co.il

תרגיל 9 - ייצוג מספרים ממשיים בזיכרון
עליכם לכתוב את ייצוג המספרים הבאים בזיכרון
כ - float, double ו-long double.

9.0 (1)

0.625 (2)

1.4 (3)

תרגיל 10 - המעבד המתמטי
סעיף א':

שיטת ניוטון-רפסון (או כלל ניוטון) היא אלגוריתם יעיל באנליזה נומרית למציאת שורשים של פונקציה ממשית כלשהי. נשתמש בשיטה זו למציאת שורש של X.

לאחר n איטרציות ניתן למצוא את $\sqrt[n]{X}$ ע"י הסדרה הבאה:

$$x_0 = X, \quad x_{n+1} = \frac{x_n}{2} + \frac{X}{2x_n}$$

ממש פרוצדורה בשפת אסמלי שתקרא מתוך תכנית C המחשבת קירוב נומרי של \sqrt{X} . הגדרת הפרוצדורה היא:

```
float my_sqrt(float x, int n);
```

קוד C הקורא לפרוצדורה הוא:

```
#include <stdio.h>

extern float my_sqrt(float x, int n);

void main()
{
    float x;
    printf("Enter x:\n");
```

לפתרון מלא בסרטון וידאו היכנסו ל- www.Gool.co.il

```
scanf("%f", &x);
printf("my_sqrt(%f) = %f\n", x, my_sqrt(x, 5));
}/* end of main */
```

סעיף ב':

יש לכתוב קוד לפתרון התרגיל עבור:

.1 double:

הגדרת הפרוצדורה היא:

```
double my_sqrtd(double x, int n);
```

.2 long double:

הגדרת הפרוצדורה היא:

```
long double my_sqrtld(long double x, int n);
```

בהצלחה !!!

לפתרון מלא בסרטון וידאו היכנסו ל- www.Gool.co.il

© GOOL כל הזכויות שמורות לאתר