

מטלה 7

למען הסר ספק, עברנו ל mips. מי שיגיש קבצי assembly של hack לא יקבל ניקוד כלל.

כווצו את כל קבצי המטלה ל zip/rar בודד ששמו מספר הזהות של המגיש.

אין הגשה בזוגות במטלה הזו!

כל סעיף הוא קובץ asm. חדש. שני הסעיפים האחרונים יהיו קבצי txt.

שימו לב: בשאלות הקשורות לזיכרון, אין צורך לאתחל ערכים בזיכרון בשביל לכתוב את הקוד. לכן, אתחול ישמש לכם לבדיקה עצמית בלבד. את הקודים יש להגיש ללא אתחול. **ירדו נקודות במקרה שהאתחול עדיין יהיה חלק מהקוד.**

יש לכתוב הערות בקוד במידה סבירה (לכל הפחות, כל שורה שלישית). קוד ללא הערות שלא ירוץ תקין לא יקבל ניקוד כלל.

בדיקת המטלה תהיה מדגמית – יבדקו 2-5 סעיפים בלבד, ושאר הסעיפים יהיו "בדיקת קיום" בלבד.

תרגילים

(1) הכניסו את מספר הזהות שלכם לאוגר \$s6. (לא להמיר מעשרוני! להכניס כאילו מספר הזהות שלכם כתב כתוב בהקסה-דצימאלי). את ספרת הביקורת (הספרה ה-LSB) יש לשים בתור ספרת ה MSB של אוגר \$t3.

(2) בנו קוד המבצע את החישוב הבא:

$$sum = \sum_{i=1}^{10} A_i \cdot B_i \cdot C_i$$

כאשר נתון

$$A = [1, 3, 5, 9, \dots, 19]$$

$$B = [1, 4, 16, 64, \dots, 2^{18} = 262144]$$

$$C = [1, -1, 1, -1, \dots, -1]$$

את התוצאה יש לשים באוגר \$s0.

(3) כתבו פונקציה רקורסיבית המממשת את הפסודו-קוד הבא:

```
Int something(int a, int b){  
    If (a ≤ 1 || b ≤ 1)  
        return max(a,b)  
    Else return (3+something(a-1,b-2))
```

(4) תרגמו את הקוד הבא מפסודו-קוד לאסמבלי.

```
Int sum(int a, int b, int c){
    If (a<b)
    {
        For (int i=0; i<=c; i++)
            a+= i;
    }
    else
    {
        a=func2(a,c);
    }
    Return a
}

Int func2(int a, int c){
    return c*2-a;
}
```

(5) תרגמו את הקוד הבא מאסמבלי של MIPS להקסה-דצימאלי.
נתונים נדרשים:

$funct(add) = 32, opcode(beq) = 4, opcode(bne) = 5,$
 $opcode(sw) = 43, opcode(lw) = 35, funct(sll) = 0, opcode(j) = 2,$
 $opcode(jal) = 3, funct(sub) = 34$

```
0x00400000: add $s0, $s1, $t0
0x00400004: bne $s0, $0, else
0x00400008: sw $t0, 0($sp)
0x0040000C: jal mult
0x00400010: lw $t0, 0($sp)
0x00400014: j skip
0x00400018: else: beq $s4, $t3, skip
0x0040001C: mult: sll $s1, $s1, 1
0x00400020: sub $t8, $s7, $t0
0x00400024: skip:
```

(6) הקוד הבא כבר עבר *Assembler*. המירו אותו חזרה לאסמבלי.
היעזרו בנתונים מהסעיף הקודם

0x00114200

0x01104820

0xad53ffc

0x152f0004

0x01766822

0x8e880004

0x08100003

0x1111fff8

טסטים למטלה:

- (1) הריצו ב *MARS* וראו את מספר הזהות שלכם באוגרים המבוקשים.
- (2) לכולם אמור לצאת אותו ערך בקוד הזה.
- (3) כתבו פונקציית *main* עם אתחול ערכים, והריצו את הפונקציה. זכרו להוציא את ה *main* לפני הגשת הקוד!
- (4) בדקו שהקוד שלכם רץ. גם כן, ניתן לאתחל ערכים כל עוד אתם זוכרים להוציא אותם לפני ההגשה.
- (5) הכניסו את הקוד ל *MARS* וקמפלו אותו. בדקו שהתרגום שלכם זהה לתרגום של *MARS*.
- (6) הכניסו את הקוד שפענחתם (צפנת פענח) ל-*MARS* וקמפלו אותו. בדקו שהתרגום של הקוד שפענחתם זהה למה שכתוב בשאלה.

בהצלחה!