**תרגיל בית 4 - יסודות המחשוב**

**שאלה 1**

בהרצאה ובתירגול למדנו את דרך הפעולה של האסמבלר ודוגמה לתכן אפשרי למימוש האסמבלר כולל מימוש של חלק מהמודולים. עליך להשלים את המימוש של האסמבלר.  
במודל בחלק שעוסק באסמבלר תמצאו פרויקט ובו מימוש של המודולים Parser, Code, SymbolTable, ודוגמה לשימוש בהם. עליכם להשלים את המודול HackAssembler.  
בתיקיה "inputs and outputs for testing" תמצאו קבצי אסמבלי וקבצי שפת מכונה תואמים ותקינים לצורך השוואת תוצאת ההרצה, עליכם להריץ את האסמבלר שאתם כותבים על קבצי שפת המכונה ולבדוק בעזרת diff את קבצי שפת המכונה שהאסמבלר שאתם כותבים מייצר לקבצי האסמבלר הנתונים.  
דרך ביצוע בדיקת התוכנית - ההתחלה של המימוש של מודול ה-HackAssembler מקבלת כפרמטר ב-argv את השם של קובץ האסמבלי לתירגום לשפת מכונה. במידה שלא מועבר פרמטר כזה כמו בהפעלה בעזרת סביבת Visual Studio ברירת המחדל היא הקובץ file.asm. כדי לבדוק את התוכנה שתכתבו החליפו קובץ זה בדוגמאות שניתנות לכם ובצעו השוואה עם קבצי שפת המכונה שניתנים לכם לשם השוואה.

יש להגיש את הקובץ HackAssembler.c שכתבתם בלבד אחרי שוידאתם שהוא מתקמפל ועובד היטב בפרויקט שקיבלתם ומתרגם את קבצי האסמבלי לדוגמה לקבצים זהים לקבצי שפת המכונה שקיבלתם.

**שאלה 2**

**הערה: תוכלו לענות על שאלה זו אחרי לימוד כתיבת קוד ב-VM וכתיבת פונקציות ב-VM.**

ממשו בשפת VM את הפונקציה SumOfSquares(n) שמקבלת מספר טבעי ומחזירה את חישוב הסדרה: . על הפונקציה לבצע את החישוב בצורה רקורסיבית, כלומר בעזרת הנוסחה:

1. כיתבו את הפונקציה הדרושה בשאלה בפסאודו קוד מותאם. כלומר כזה שמכיל פקודות שקיימות בשפת VM (מומלץ לכתוב ראשית את הקוד בשפת C פשוטה ואח"כ לתרגם את הקוד לפסאודו קוד מותאם).

2. תרגמו את קוד הפסאודו קוד המותאם שכתבתם לקוד VM, כך ששורות הפסאודו קוד יופיעו כהערות והתרגום של כל שורה תופיע אחריה. הניחו שהקוד הוא חלק מהקובץ q3.vm. אין צורך לכתוב קריאה לפונקציה.

**שאלה 3**

**לשאלה זו מצורף פיתרון והיא לתירגול ואין צורך להגישה.**

כיצד תתורגם הפקודה sub **ע"י ה VM translator** לאסמבלי של ה HACK?

**פיתרון שאלה 3**

//SP--

@SP

M=M-1

//D=\*SP

A=M

D=M

//SP--

@SP

M=M-1

//D=D-\*SP

A=M

D=D-M

//\*SP=D

@SP

A=M

M=D

//SP++

M=M+1

**שאלה 4**

**לשאלה זו מצורף פיתרון והיא לתירגול ואין צורך להגישה.**

א. מה יהיה הקידוד בשפת מכונה של הפקודות הבאות בהנחה שהמשתנה x הוא המשתנה הססטטי הראשון שהוגדר בתוכנית:

@x

M=M-D

ב. צייר איך תראה המחסנית בעת ביצוע הפונקציה הבאה בנקודה המסומנת. הנח שהפונקציה נכתבה לפי הקונוונציה המקובלת ב-C כפי שנלמדה בכיתה שכוללת שמירת BP במחסנית והקצאת משתנים מקומיים במחסנית. צייר את כל האיברים במחסנית שחשובים לפעולתה של הפונקציה. כתוב עבור כל איבר מה הוא מיצג ואת הערך שלו אם הוא ידוע. סמן על מה מצביע BP ועל מה מצביע SP בנקודה הזו.

הפונקציה:

int func(int a, int b, int\* c, int d, int e) {

int f=11, g=12, h=13, i=14;

//<------draw the stuck in this point of execution

...

return h;

}

הקריאה לפונקציה:

a = func(1, 2, &x, 3, 4);

ג. תרגם את הקוד של הבנאי של המחלקה הבאה שכתוב בשפת Jack לשפת VM:

class ClassA {

field int a;

field int b;

field int c;

constructor ClassA new(int aa, int bb, int cc) {

var int d;

d = bb + cc;

let a = aa;

let b = bb;

let c = d;

return this;

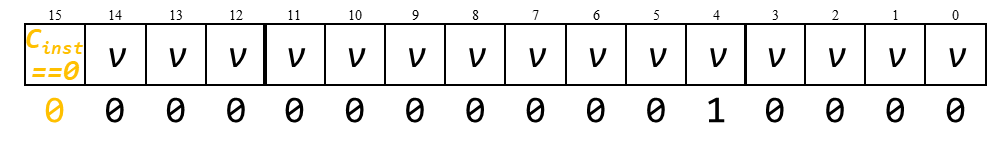
}

...

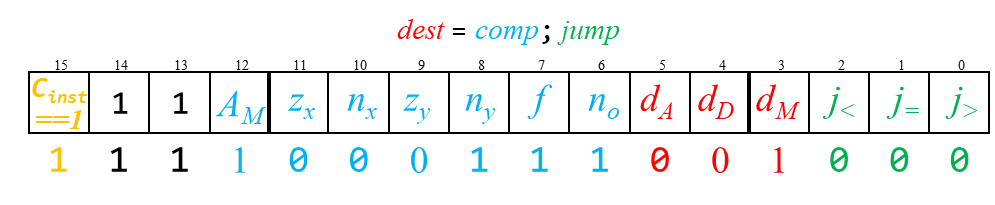
}

**פיתרון שאלה 4**

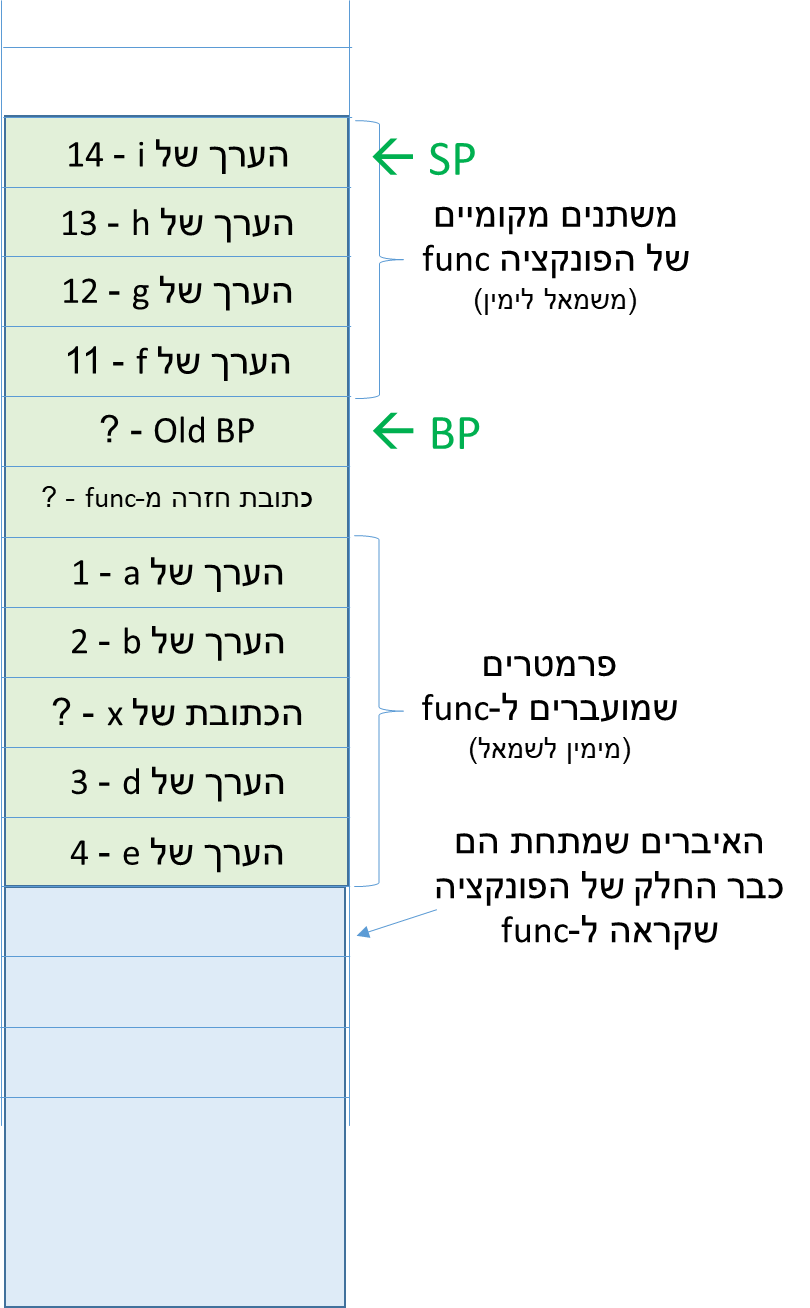
א. הפקודה הראשונה היא מסוג A והקידוד שלה הוא:



הפקודה השניה היא פקודה מסוג C והקידוד שלה הוא:



ב. המחסנית תראה כך:



ג.

//constructor ClassA new(int aa, int bb, int cc) {

//var int d;

function ClassA.new 0

//argument 0 – aa, argument 1 – bb, argument 2 – cc

//local 0 – d

//this 0 – a, this 1 – b, this 2 – c

//allocating object memory

push constant 3

call Memory.alloc 1

pop pointer 0

//d = bb + cc;

push argument 1

push argument 2

add

pop local 0

//let a = aa;

push argument 0

pop this 0

//let b = bb;

push argument 1

pop this 1

//let c = d;

push local 0

pop this 2

//return this;

push pointer 0

return

**הוראות הגשה:**

ההגשה **בזוגות בלבד**. ההגשה במודל לפי התאריך המעודכן בתיבת ההגשה. יש להגיש קובץ hw.zip אשר מכיל:

1. קובץ HackAssembler.c עם פתרון שאלה 1.
2. קובץ pow.vm וקובץ powFunc.vm עם פיתרון שאלה 2.

**שאלות נא להפנות למרצה האחראית על התרגיל רן דרור במייל** [**rand@braude.ac.il**](mailto:rand@braude.ac.il)**.**

**בהצלחה מכל צוות הקורס!**