

מורים מובילים תשע"ח - אתגרים לתלמידים מצטיינים

מסמך מתעדכן - גרסה 1.0 - מאי 2018

מבוא: המסמך מכיל ריכוז של מטלות חשיבה וחקר לתלמידי תיכון מצטיינים או כשעורי העשרה לתלמידי הנדסת תוכנה, לביצוע במעבדת מחשבים, ליחידים או כעבודה בקבוצות. מהניסיון ומסקרים, אילו שעורים מוצלחים בהם התלמידים נהנים (בפרט Nand2Tetris).

תוכן

1	מצילים את הרשת
1	מהלך השיעור
2	פתרון
4	טיטניק - למידה מאסונות
6	מהלך השיעור ודרך הפתרון
6	שיעור המשך אפשרי
6	Tetris2Nand - יסודות מדעי המחשב
7	מהלך שיעור ראשון - רכיבים בסיסיים
7	מהלך שיעור שני - בניית יחידת ALU שב CPU
7	מהלך שיעור שלישי - בניית רכיבי זיכרון - רגיסטרים וזיכרון RAM
8	פתרון חלקי



מצילים את הרשת

- קרידט: תוכנית מגשימים
- אוכלוסיית יעד: תלמידים מצטיינים או הלומדים הנדסת תוכנה
- תחום: הגנת סייבר
- ידע ומיומנות נרכשות: מבוא לסייבר, למידת שפת פקודות חדשה ויישומה
- ידע מוקדם הנדרש ע"מ לפתור: אין.
- קבצים:
 - [מצגת תיאור הבעיה לתלמידים](#)
 - [האמולטור](#)

מהלך השיעור

- מומלץ להציג בפני הכיתה את הבעיה המוצגת במצגת המצורפת.
- מומלץ לשלוח לתלמידים
 - קישור למצגת - כך שהפקודות תהינה זמניה

- קובץ אמולטור - התלמידים צריכים לפתוח את הקובץ הדחוס על הדיסק המקומי ולהריץ מהספרייה את הקובץ network.exe. אמורים לקבל מסך דמוי shell. בו יש להזין את הפקודות המותרות. הפקודה help מציגה את רשימת הפקודות.

```

C:\WINDOWS\system32\cmd.exe - network.exe
E:\WUTemp\network>network.exe

LocalHost# help

Documented commands (type help <topic>):
=====
Back          Exit          GetRemoteHostOS      ShowPayload
ConnectHost  GetFileList  GetRemoteHostServices ShowPayloads
DownloadFile  GetNeighbors  Rescue

Undocumented commands:
=====
help

LocalHost# help GetNeighbors
Shows all current host neighbors
LocalHost#

```

- התלמידים נדרשים להבין לבד מה הן דרכי הפעולה על בסיס הפעולות האפשרויות, כמפורט במצגת.

- המטרה למצוא את הרעים - קובץ המכיל את התמונה של שרדר



פתרון

- בפתרון על התלמידים להתחיל מהמחשב הנוכחי:
 - i. לסרוק אילו מחשבים מחוברים אלי ברשת

```

LocalHost# GetNeighbors
MagshimimForum
MagshimimForumFiles

```

- ii. לבדוק מה מאפיין את אותם המחשבים - סוג מערכת הפעלה באותם מחשבים או את השירות שרץ עליהן (שרת דואר אלקטרוני, שרת מסד נתונים וכו')

```
LocalHost# GetRemoteHostOS MagshimimForum
MagshimimForum's Operating System is: Windows 2008 Server
LocalHost# GetRemoteHostServices MagshimimForum
MagshimimForum's open services are:
HTTP WEB Server
```

.iii קיים מאגר של חולשות דרכן ניתן לפרוץ למחשב ולהשיג את קוד המשתמש והססמה של המחשב (המאגר מוצג ע"י אחת מהפקודות). כל חולשה כזו מאופיינת ע"י מערכת הפעלה או שירות.

```
LocalHost# ShowPayloads
UID: 1
Name: MS Windows SMB Authentication Remote Exploit
Description: Exploits vulnerability in Microsoft Server Message Block (SMB) Protocol. The vulnerability works on All of the Windows XP Versions (Including Service packs), and Above (98, 95..). It allows remote code execution on affected systems. An attacker who successfully exploited this vulnerability could install programs, view, change, or delete data or create new accounts with full user rights.
UID: 2
Name: Windows Routing and Remote Access Service RPC Buffer Overflow
Description: A vulnerability in Windows 2003 Server's Routing and Remote Access Service. A remote user can execute arbitrary code on the target system. A remote user can send specially crafted RPC data to the routing and remote access service to trigger a buffer overflow and execute arbitrary code on the target system. The code will run with the privileges of the target service.
UID: 3
Name: VMBus Persistent DoS Vulnerability
Description: Hyper-V in Microsoft Windows Server 2008 Gold, SP2, R2, and R2 SP1 allows guest OS users to cause a denial of service (host OS infinite loop) via malformed machine instructions in a VMBus packet, aka "VMBus Persistent DoS Vulnerability."
UID: 4
Name: Fax Cover Page Use After Free Vulnerability
```

.....
.iv קבלת קוד המשתמש ע"י פריצה למחשב, בעזרת פקודה המקבלת שם מחשב / שרת וסוג החולשה שיש להפעיל על מנת לתקוף אותו.

```
LocalHost# Rescue MagshimimForum 3
[██████████] 100% Trying to Rescue: Complete!
Rescuing remote host went successfully!
MagshimimForum's credentials:
Username: forum_admin
Password: BestForumEver
```

.v התחברות למחשב המרוחק.

```
LocalHost# ConnectHost MagshimimForum forum_admin BestForumEver
[ ] 100% Trying to connect: Complete!
You have been connected successfully to MagshimimForum
```

- .vi בדיקת רשימת הקבצים שיושבים עליו.
- .vii הורדת הקבצים שבו למחשב של התלמיד ובדיקתם.
- .viii במידה ולא נמצא הפתרון (קובץ בשם GoodJobGuys.jpg עם התמונה הנ"ל), יש לחזור לסעיף הראשון ולהמשיך לסרוק את המחשבים ברשת.



2. טיטניק - למידה מאסונות

- קרדיט: אתר www.kaggle.com
- תחום: למידת מכונה ואינטלגנציה מלאכותית
- אוכלוסיית יעד: תלמידים מצטיינים או הלומדים הנדסת תוכנה
- ידע ומיומנות נרכשות:
 - הבסיס של למידת מכונות (Machine Learning) ו-AI
 - מיון למחלקות לפי הסתברות
 - בסיס למחקר עבור פרויקט בהנדסת תוכנה.
- ידע מוקדם נדרש: הגדרה להסתברות של מאורע, כנלמד במתמטיקה 5 יח"ל בכיתה י"א.
 - הסתברות היא ביטוי מספרי למידת הסבירות שמאורע מסוים יתרחש. ההסתברות של מאורע יכולה לקבל ערך מספרי שבין 0 ל-1. מאורע בלתי אפשרי הוא בעל הסתברות 0, ומאורע ודאי הוא בעל הסתברות 1. ההסתברות היא מושג יסודי במתמטיקה.
 - לדוגמה: כאשר אנו מטילים מטבע, קל לומר באופן אינטואיטיבי שההסתברות שהוא יפול על "עץ" היא 50%. כאשר אנו מטילים מטבע 10 פעמים, נוכל לצפות שב-5 מקרים ייצא "עץ" וב-5 מקרים ייצא "פלי", אך ברור שאין כל ודאות בכך.
- רקע:
 - טביעתה של הטיטאניק שהיא אחת מספינות הטרופות הידועה ביותר לשמצה בהיסטוריה. אפריל 15, 1912, בעת הפלגת הבכורה שלה, הטיטניק טבעה לאחר התנגשות עם קרחון, 1502 נהרגו מחוץ 2224 הנוסעים ואנשי הצוות. הטרגדיה הסנסציונית הדהימה את הקהילה הבינלאומית והובילה לשיפור תקנות הבטיחות בספינות.
 - אחת הסיבות שטביעת הספינה גרמה לאבדות רבות בנפש, היה כי לא היו מספיק סירות הצלה עבור הנוסעים ואנשי הצוות. למרות מעורבות של אלמנט המזל, מספר קבוצות של אנשים היו בסבירות גבוהה יותר לשרוד מאשר אחרים, כגון נשים, ילדים והמעמד העליון.
 - באתגר זה, אנו מבקשים ממך להשלים את הניתוח של איזה מיני אנשים היה סביר יותר שישרדו. בפרט, אנו מבקשים ממך להשתמש בכלים של תורת הלמידה לחזות אילו נוסעים שרדו את הטרגדיה.
- כלי עבודה:
 - מומלץ - גיליון אקסל
 - אפשרות - לכתוב קוד בשפת תכנות
- מטרה
 - השתתפות בתחרות: לחזות אם נוסע שרד טביעתה של הטיטאניק או לא. עבור כל מספר זיהוי של נוסע (PassengerId) בקובץ הבדיקה, עליך לנבא ערך 0 או 1 עבור המשתנה המייצג האם שרד או לא.

○ הציון שלכם בתחרות הוא האחוז של הנוסעים שעבורם הצלחתם לחזות בצורה נכונה (רמת ה"דיוק" שלכם).

○ על מנת לקבל ציון, יש להעלות קובץ CSV, בפורמט שמוגדר באתר.

● מקורות הנתונים:

○ אתר האתגר - <https://www.kaggle.com/c/titanic>

○ קובץ האימון ([train.csv](#)) - (משמש לבניית מודל ללימוד המכונה. הקובץ מכיל, עבור כל נוסע, את תכונותיו וגם את התוצאה - האם ניצל (המכונה גם "בסיס האמת"). המודל של התלמידים יתבסס על "תכונות" כמו מין והמחלקה באוניה של הנוסעים. ניתן גם לפעולות על התכונות, כדי ליצור תכונות חדשות.

Variable	Definition	Key / Notes
survival	Survival	0 = No, 1 = Yes
pclass	Ticket class	1 = 1st = Upper, 2 = 2nd = Middle, 3 = 3rd = Lower
sex	Sex	
Age	Age in years	Age is fractional if less than 1. If the age is estimated, is it in the form of xx.5
sibsp	# of siblings / spouses aboard the Titanic	The dataset defines family relations in this way... Sibling = brother, sister, stepbrother, stepsister Spouse = husband, wife (mistresses and fiancés were ignored)
parch	# of parents / children aboard the Titanic	The dataset defines family relations in this way... Parent = mother, father Child = daughter, son, stepdaughter, stepson Some children travelled only with a nanny, therefore parch=0 for them.
ticket	Ticket number	
fare	Passenger fare	
cabin	Cabin number	
embarked	Port of Embarkation	C = Cherbourg, Q = Queenstown, S = Southampton

○ קובץ המבחן ([test.csv](#)) - (משמש לבחינה עד כמה המודל שהתלמידים פיתחו פועל על נתוני נוסעים, ללא המידע מה קרה לנוסעים. התפקיד של התלמידים לחזות את התוצאות הללו. יש להפעיל את המודל

שבבנה ע"ב נתוני האימון, על כל נוסע בקובץ המבחן, על מנת לחזות אם הוא שרד את הטביעה של הטיטניק או לא.

- דוגמה לפורמט קובץ התוצאה ([gender_submission.csv](#)) - (מכיל את התוצאה של הפעלת המודל: קוד נוסע (הנוסעים מסודרים בסדר כלשהו) והאם שרד (0 - נפטר, 1 - שרד). יש לשדר את התוצאה לאתר הנ"ל ע"מ לקבל ציון.

מהלך השיעור ודרך הפתרון

- התחלה: התלמידים מחפשים באינטרנט מה ההגדרה והשימוש ב"למידת מכונה" ומתקיים על כך דיון בהובלת המורה. דגשים: התחום עוסק בפיתוח אלגוריתמים המיועדים לאפשר למחשב ללמוד מתוך דוגמאות, מבלי להיות מתוכנתים באופן ספציפי. תת-תחום במדעי המחשב ובבינה מלאכותית, המשיק לתחומי הסטטיסטיקה, לדוגמה: עבור זיהוי תווים, מערכות זיהוי דיבור. טיפול ממוחשב בנתונים מן העולם האמיתי לפתרון בעיה מסוימת.
- שלב א': ניסיון לחזות האם נוסע ניצל לפי מין. מצופה שהתלמידים יחשבו בגיליון האלקטרוני את ההסתברות להישרדות של כל אחד מהמינים ויתנו תחזית בהתאם
- שלב ב': ניסיון לבחון אילו משתנים משפיעים על הישרדות הנוסע ובניית עץ הסתברות: מה ההסתברות לנוסע שהוא גם ממין מסוים וגם בגיל מסוים וגם ישן במחלקה מסוימת וכו'.

שיעור המשך אפשרי

רשת נירונים לזיהוי ספרות, ע"י מעבר מודרך בכיתה על הסרטונים הבאים:

- [what *is* a Neural Network? | Chapter 1, deep learning](#)
- [Gradient descent, how neural networks learn | Chapter 2, deep learning](#)
- [What is backpropagation really doing? | Chapter 3, deep learning](#)




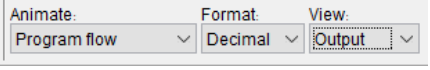
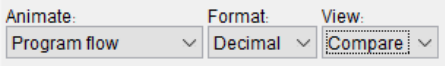


Nand2Tetris.3 - יסודות מדעי המחשב

- קרדיט: האתגר מתבסס על חומרים מקורס אקדמי, רחב תפוצה, בשם Nand2Tetris, שפותח ע"י פרופ' נועם ניסן מהאוניברסיטה העברית ופרופ' שמעון שוקן מהמרכז הבין תחומי בהרצליה. הקורס מוגדר כבסיסי וידע בתכנות הינו תנאי הקדם היחיד עבורו. יתרונו בשילוב הנושאים המרכזיים של תחום מדעי המחשב, באופן בו התלמיד מבין את ההקשר בין כל הנושאים, [משערים לוגיים בחומרה](#), [דרך אסמבלי](#), [מהדר](#), [מערכת הפעלה](#) [ועד לפיתוח משחק שרץ מעל המחשב שהתלמיד בנה](#). כל חומרי הלימוד והתוכנות הרלוונטיות נמצאות באתר הקורס <http://nand2tetris.org/>
- מהות: בניית מחשב מודרני מהיסודות מדעי המחשב, הנלמדים בתיכון ו/או בתואר הראשון.
- תחום ידע: מערכות סיפרתיות (התחום הראשון בקורס. ניתן להרחיב או לצמצם את היקף ההשקעה, בהתאם לרצון התלמידים)
- אוכלוסיית יעד: תלמידים מצטיינים או הלומדים הנדסת תוכנה
- ידע ומיומנות נרכשות:
 - מתמטיקה בולאנית
 - מערכות ספרתיות ותכן לוגי
 - מבנה המחשב
- ידע מוקדם נדרש: אין

- הנחיות התקנה של קבצי התרגול ולסימולטור: [מופיעות באתר](#). ע"מ להריץ את היישומים, יש להתקין JRE - [Java Run-time Environment מהאתר הרשמי](#).

מהלך שיעור ראשון - רכיבים בסיסיים

- **מצגת 01 לוגיקה בולאנית** - מתמטיקה בוליאנית, שערים לוגיים, סביבת הסימולטור להקמת רכיבים ספרתיים, בניית רכיבים ספרתיים בעזרת שפת HAL - Hardware Description Language
- אופן בניית הרכיבים ובדיקתם ע"י התלמידים, בעזרת הסימולטור:
 - עריכת קובץ בשפת hdl (עם סיומת hdl) וכתובת הקוד להגדרת הרכיב (הקשר בין הקלט לפלט) - בתוך תת ספרייה `01\nand2tetris\projects`
 - כניסה לסימולטור ע"י הרצת `nand2tetris\tools\HardwareSimulator.bat` (או `nand2tetris\tools\HardwareSimulator.sh` לעובדים בלינוקס)
 - טעינת התסריט לפעולת הרכיב, שנכתב בשפת hdl (בעזרת הכפתור  בבר העליון). הפעולה לא תתבצע במידה וקיימת שגיאה תחבירית בקובץ זה.
 - טעינת תסריט בדיקות קיים - קובץ ששמו מורכב משם הרכיב והסיומת `tst` (בעזרת הכפתור  בבר העליון).
 - הרצת תסריט הבדיקות (בעזרת הכפתורים  בבר העליון).
 - הצגת הפלט (טבלת אמת) שהתקבל מהרכיב - 
 - הצגת הפלט הנדרש, להשוואה -  במידה והפלט מהסעיף הקודם זהה לפלט שהתקבל כאן, הרכיב נבנה באופן נכון.
 - ניתן להיעזר [במדריך למשתמש](#) לתוכנה, עפ"י הצורך.
- סדר בניית רכיבים בפרק הראשון (חשיבות לסדר, מאחר וחלק מהרכיבים משתמשים ברכיבים שכבר נבנו ע"י התלמיד):
 - (רכיב בסיסי קיים) NAnd
 - Not, And, Or, Xor, Mux, DMux, Not16, And16, Or16, Mux16, Or8way, Mux4way16, Mux8way16, DMux4way, DMux8way
- תלמידים מתקדמים יכולים להמשיך לבניית רכיבים המפורטים בשיעור השני.

מהלך שיעור שני - בניית יחידת ALU שב CPU

- כללי: בפרק זה, יתרון לתלמידים שלמדו אסמבלי.
- **מצגת 02 מתמטיקה בולאנית** - בסיסי ספירה, פעולות חשבון בבסיס 2, שיטת המשלים ל-2 למציאת המספר הנגדי, בניית רכיב המחבר מספרים שלמים (16 ביט), בניית ALU - היחידה האריתמטית לוגית שב-CPU - שמקבלת שני רגיסטרים ומבצעת חיבור, חיסור, איפוס, מציאת נגדי, וגם / או וכו' (ללא כפל / חילוק ופעולות מורכבות, שיבוצעו במחשב זה ברמת התוכנה).
- סדר בניית רכיבים - פרק שני:

HalfAdder, FullAdder, Add16, ALU, Inc16

מהלך שיעור שלישי - בניית רכיבי זיכרון - רגיסטרים וזיכרון RAM

- **מצגת 03 - לוגיקה סדרתית** - שעון טיק טוק, שערי פליפ פלופ, מימוש זיכרון בצורה רקורסיבית, מימוש מונה
 - סדר בניית רכיבים - פרק שלישי:
- רכיב בסיסי קיים עבור כל הרכיבים הסדרתיים) DFF
 Bit, Register, RAM8, RAM64, RAM512, RAM4K, RAM16K, PC (Program counter)

פתרון חלקי

```
/**
 * Not gate:
 * out = not in
 */

CHIP Not {
  IN in;
  OUT out;

  PARTS:
  // Put your code here:
  Nand(a=in, b=in, out=out);
}
```

```
/**
 * And gate:
 * out = 1 if (a == 1 and b == 1)
 *   0 otherwise
 */
```

```
CHIP And {
  IN a, b;
  OUT out;

  PARTS:
  // Put your code here:
  Nand(a=a, b=b, out=x);
  Not(in=x, out=out);
}
```

```
/**
 * Or gate:
 * out = 1 if (a == 1 or b == 1)
 *   0 otherwise
 */
```

```
CHIP Or {
  IN a, b;
  OUT out;

  PARTS:
  // Put your code here:
  Not(in=a, out=na);
```



```
    Not(in=b, out=nb);
    Nand(a=na, b=nb, out=out);
}
```

```
/**
 * Exclusive-or gate:
 * out = not (a == b)
 */
```

```
CHIP Xor {
```

```
    IN a, b;
    OUT out;
```

```
    PARTS:
```

```
    // Put your code here:
```

```
        Not(in=a, out=na);
        Not(in=b, out=nb);
        And(a=a, b=nb, out=anb);
        And(a=na, b=b, out=nab);
        Or(a=anb, b=nab, out=out);
```

```
}
```

```
/**
 * Multiplexor:
 * out = a if sel == 0
 *     b otherwise
 */
```

```
CHIP Mux {
```

```
    IN a, b, sel;
    OUT out;
```

```
    PARTS:
```

```
    // Put your code here:
```

```
        Not(in=sel, out=n sel);
        And(a=n sel, b=a, out=sel0);
        And(a=sel, b=b, out=sel1);
        Or(a=sel0, b=sel1, out=out);
```

```
}
```

```
/**
 * Demultiplexor:
 * {a, b} = {in, 0} if sel == 0
 *         {0, in} if sel == 1
 */
```

```
CHIP DMux {
```

```
    IN in, sel;
```

OUT a, b;

PARTS:

// Put your code here:

```
Not(in=sel, out=nsel);
And(a=nsel, b=in, out=a);
And(a=sel, b=in, out=b);
```

}

/**

* 16-bit Not:

* for i=0..15: out[i] = not in[i]

*/

CHIP Not16 {

IN in[16];

OUT out[16];

PARTS:

// Put your code here:

```
Not(in=in[0], out=out[0]);
Not(in=in[1], out=out[1]);
Not(in=in[2], out=out[2]);
Not(in=in[3], out=out[3]);
Not(in=in[4], out=out[4]);
Not(in=in[5], out=out[5]);
Not(in=in[6], out=out[6]);
Not(in=in[7], out=out[7]);
Not(in=in[8], out=out[8]);
Not(in=in[9], out=out[9]);
Not(in=in[10], out=out[10]);
Not(in=in[11], out=out[11]);
Not(in=in[12], out=out[12]);
Not(in=in[13], out=out[13]);
Not(in=in[14], out=out[14]);
Not(in=in[15], out=out[15]);
```

}
