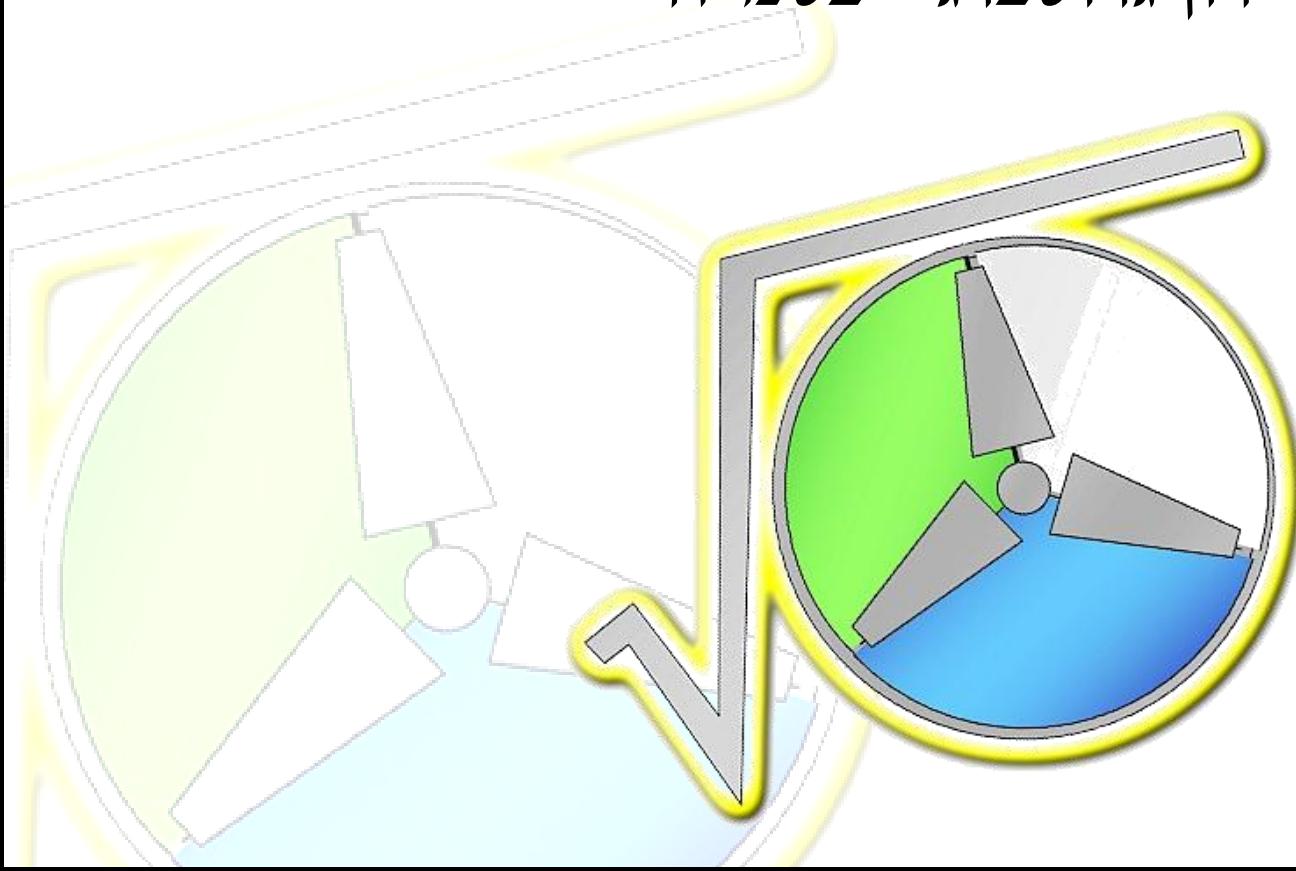


בסמ"ח – סוגיות הדרכה במחשבים

רון גרוסברג – בסמ"ח



עוד לפני ההקדמה...

- מי זה האיש החייל שעומד מולכם?
- מהו בית הספר למקצועות המחשב?
- מה עושים שם? איזה קורסים?
- קורס תכנות חדש / קורס טכ"מ חדש
- מהו ממר"ס?
- מה תפקידי בכוח?

איך בונים קורס?

- מודל תל"מ – תכנון לפי מצבים.
- קבלת הדרישות ממפקדי יחידות המחשב.
- בניית מרכיבי התפקיד של תוכניתן ע"פ הדרישות.
- בניית פרויקט הסיכום – מבחן הסיכום של הקורס
- בניית אבני דרך – מבחני ביניים.
- החלטה על חלוקה למקצועות שונים.
- בניית המקצועות ע"פ מבחני הביניים.

קורס תכנות החדש

- קורס שאורכו כחצי שנה.
- בנוי משני חלקים – חלק בסיסי ומתקדם.
- החלק הבסיסי מכשיר את האדם להיות תוכניתן.
- החלק המתקדם מפתח את התוכניתן בדברים מתקדמים – החל ממקצועות מתקדמים (DB מול א"ק) ועד מיומנויות רכות (עבודה בצוות וכו').

קורס תכנות לעתודאי טכ"מ

- מסתמך על כך שחלק אי של הקורס נלמד בבתי הספר.
- מסתמך על הנסיון שרוכשים התלמידים בכדי לקצר את זמן הלימוד של נושאים.
- אורך הקורס – 9 שבועות (בקרוב 12...).

מאפייני עבודת התוכניתן במהלך פיתוח פרוייקט תוכנה

– בכל רגע נתון, משתתף התוכניתן בפיתוח של פרוייקט תוכנה אחד בלבד.

– במהלך עבודתו בפרוייקט, התוכניתן מתבסס על תיק עיצוב ותוכנית עבודה (מוגדרת ע"י ראש הצוות ו/או המעצב).
אסור לו לסטות מהעיצוב או מתוכנית העבודה ללא אישור מהעצב/הרש"ץ.

– במקרים רבים, התוכניתן לא מכיר את סביבת העבודה שבה יעבוד בפרוייקט ועליו ללמוד אותה לפני תחילת הפיתוח, ע"י יציאה לקורסים, קריאת ספרות מקצועית (בעיקר אנגלית), תרגול וחניכה.

מאפייני עבודת התוכניתן במהלך פיתוח פרוייקט תוכנה (המשך)

– יחידת העבודה הבסיסית של התוכניתן - המודול. התוכניתן כותב מודולים באופן עצמאי.

– התוכניתן אחראי לריצה של תוכניותיו כחלק מהמערכת כולה, ועל כן עליו לבצע אינטגרציה של המערכת בין מודולים שלו למודולים של תוכניתנים אחרים.

מאפייני עבודת התוכניתן במהלך פיתוח פרוייקט תוכנה (המשך)

- במהלך פרוייקט, התוכניתן עשוי לקבל שינויים בעיצוב. שינויים אלה יכולים להיות ביוזמת התוכניתן או ביוזמת צרכן המערכת. בכל מקרה יגיעו השינויים אל התוכניתן דרך ראש הצוות/המעצב.
- עבודת התוכניתן כוללת ביצוע תיקונים במודולים שנכתבו ע"י תוכניתנים אחרים.

מאפייני עבודת התוכניתן במהלך פיתוח פרוייקט תוכנה (המשך)

- באחריות התוכניתן לחפש שגיאות בתוכניותיו ולתקנן.
- התוכניתן עובד בצוות. הוא צריך לדעת לתקשר עם שאר אנשי הצוות, לבצע מטלות צותיות מדי פעם ולעבוד לפי סטנדרטים צוותיים.
- לתכנות נלווה תהליך עבודה מסודר (מתודולוגיות, סטנדרטים, נהלי פיתוח) שהוא מסובך פחות מעצם התכנות, ופחות קריטי להצלחת התוכניתן.
- התוכניתן אחראי על התקנת המחשב שלו וסביבת העבודה בה הוא עובד.
- במרבית המקרים התוכניתן מבצע עבודת תמיכה והטמעה למערכת אותה הוא כותב.

מרכיבי התפקיד – מיומנויות טכניות

- פתרון בעיה בעזרת תכנות מבני ו-OOP
- כתיבת תוכניות המשתמשות בנתונים חיצוניים
- פיתוח באמצעות מחוללי יישומים ומחוללי דוחות
- ביצוע שינויים והכרת מערכת קיימת (תחזוקה)
- כתיבת תוכניות בסביבה מורכבת (Web, Client Server, nTier וכו')
- פיתוח של מערכות קטנות מא' – ת' (ללא צורך בעיצוב מוכן)

מרכיבי התפקיד – מיומנויות טכניות

- יכולת תכנות במגוון סביבות ותנאים
- הנדסת תוכנה
- עקרונות הנדסת תוכנה (כולל נושאים כמו בקרת תצורה)
- יכולת עבודה עם תיק עיצוב, בעיקר תיקים שאינם מושלמים
- תכנון וביצוע Unit Testing
- ממשק אדם מכונה

מרכיבי התפקיד – מיומנויות טכניות

- נושאים בתחום התשתיות
- הכרת מושגי יסוד בעולם המחשוב
- מודעות לאבטחת מידע
- מערכת הפעלה כולל מרובת תהליכים
- תפעול המחשב
- תקשורת נתונים

מרכיבי התפקיד – מיומנויות טכניות

- שפת סף
- תיאוריה במדעי המחשב
- מבני נתונים
- מושגי יסוד באלגוריתמיקה תאורטית (סדר גודל, סיבוכיות וכו')
- ארגון קבצים ומסדי נתונים

מרכיבי התפקיד – מיומנויות אישיות

- יכולת לימוד עצמי של כלים, שפות, סביבות עבודה ומתודולוגיות
- עצמאות בעבודה
- השתלבות במערכת הצבאית
- אמינות
- עבודה בצוות
- עבודה תחת לחץ זמן ועומס מטלות
- שירות ללקוח
- "ראש גדול" (אחריות, יוזמה, סקרנות, מצויינות)

צירי סיבוך

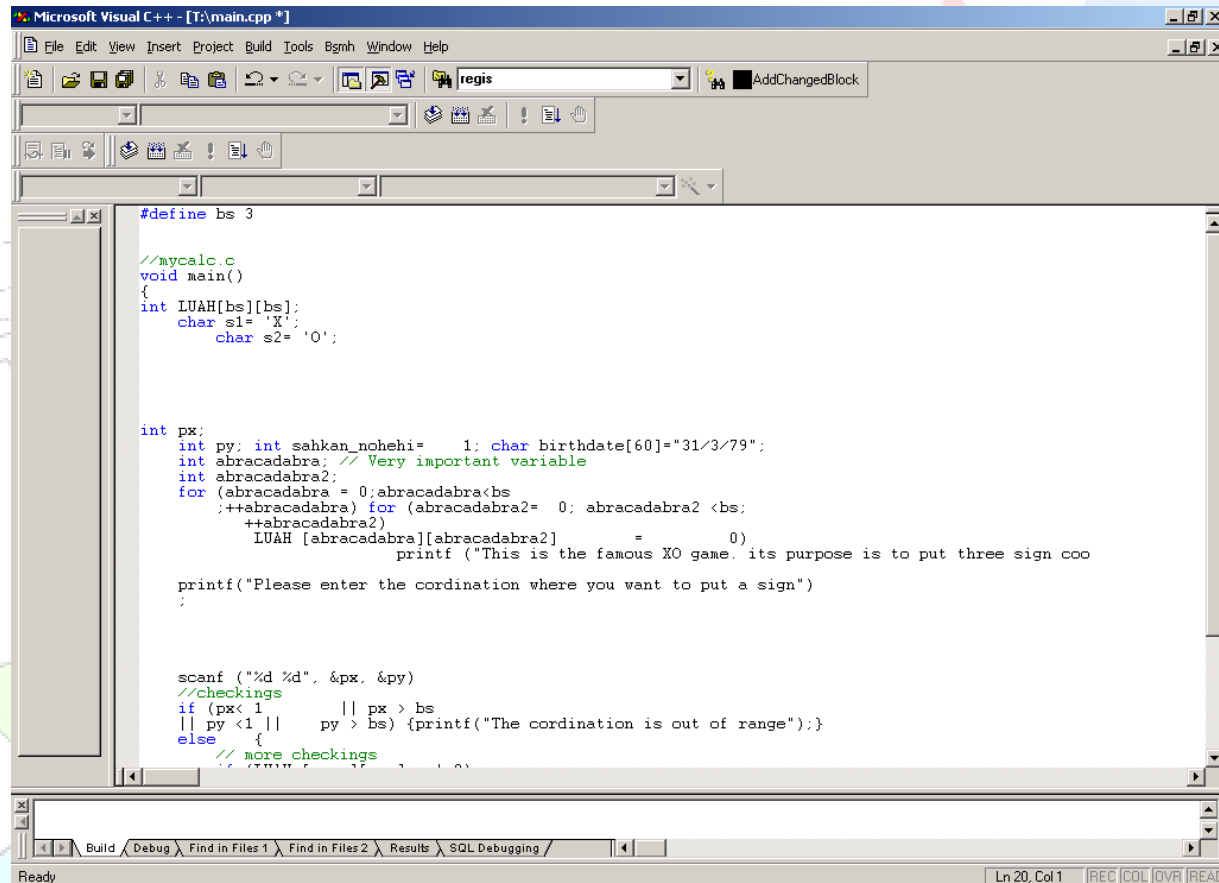
- צירי הסיבוך נגזרים ממרכיבי התפקיד של התוכנית.
- במשך הקורס מעלים את הדרישות מהחניכים בצורה מבוקרת.
- ההתקדמות בצירי הסיבוך היא מקבילית.

דוגמא – עבודה בצוות

- עובד לבד
- מדריך צמוד
- זמן קצוב
- מתכנן אלגוריתם מראש
- מדריך דוגם מדי זמן
- ביצוע עבודת תחזוקה על מערכת קיימת
- שימוש בעיצוב מכלי CASE
- עבודת צוות
- דרישות משתנות
- עיצוב משתנה
- ניהול זמן עצמאי
- רשי"צ לא זמין
- צוות משתנה
- לוי"ז משתנה

סטנדרטים - דוגמא

• עבודה ע"פ סטנדרטים



```
#define bs 3

//mycalc.c
void main()
{
    int LUAH[bs][bs];
    char s1= 'X';
    char s2= '0';

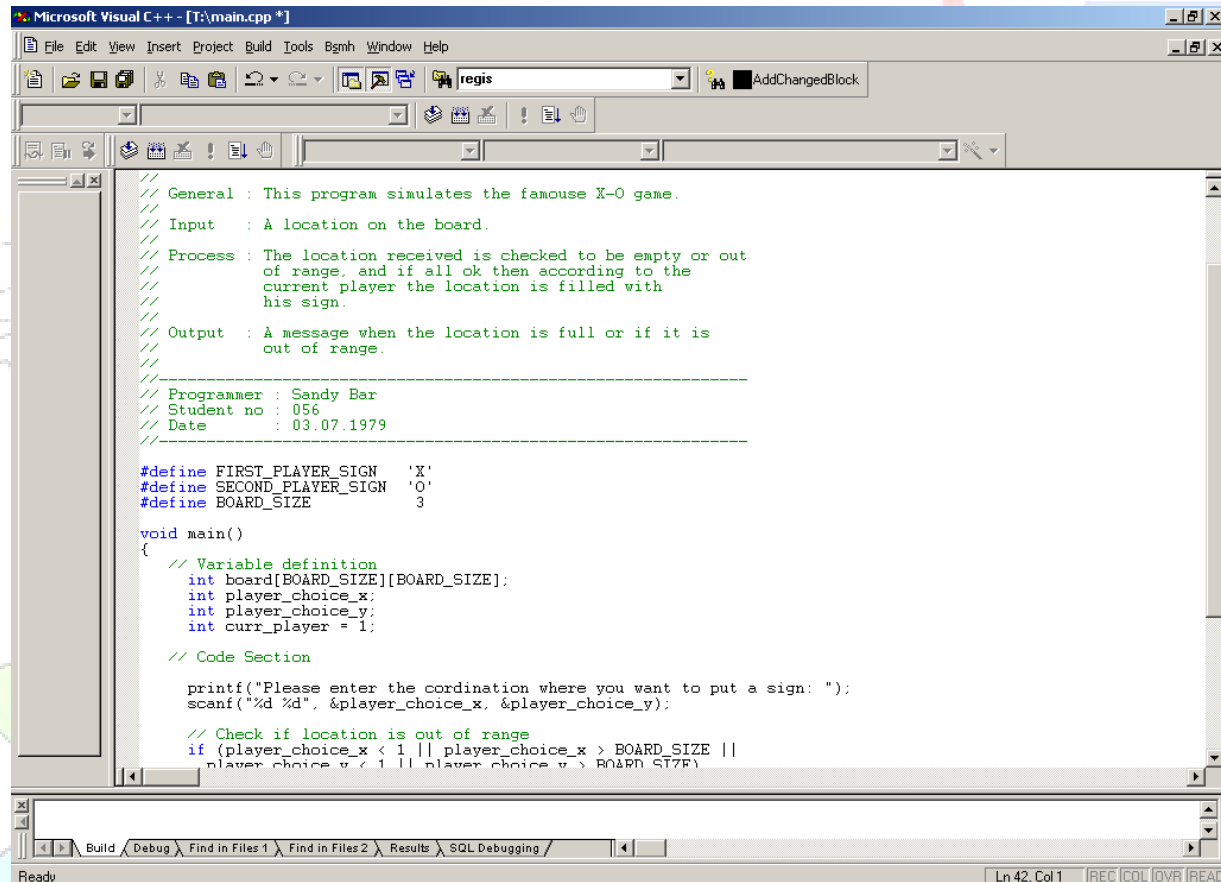
    int px;
    int py; int sahkan_nohehi= 1; char birthdate[60]="31/3/79";
    int abracadabra; // Very important variable
    int abracadabra2;
    for (abracadabra = 0; abracadabra < bs;
        ++abracadabra) for (abracadabra2= 0; abracadabra2 < bs;
            ++abracadabra2)
        LUAH [abracadabra][abracadabra2] = 0;
    printf ("This is the famous XO game. its purpose is to put three sign coo

    printf("Please enter the cordination where you want to put a sign")
    ;

    scanf ("%d %d", &px, &py)
    //checkings
    if (px < 1 || px > bs
        || py < 1 || py > bs) {printf("The cordination is out of range");}
    else {
        // more checkings
        // (many more checkings)
    }
}
```

סטנדרטים – דוגמא

• עבודה ע"פ סטנדרטים



```
Microsoft Visual C++ - [T:\main.cpp *]
File Edit View Insert Project Build Tools Bgnh Window Help
regis AddChangedBlock

// General : This program simulates the famous X-O game.
// Input  : A location on the board.
// Process : The location received is checked to be empty or out
//           of range, and if all ok then according to the
//           current player the location is filled with
//           his sign.
// Output  : A message when the location is full or if it is
//           out of range.
//-----
// Programmer : Sandy Bar
// Student no  : 056
// Date       : 03.07.1979
//-----

#define FIRST_PLAYER_SIGN 'X'
#define SECOND_PLAYER_SIGN 'O'
#define BOARD_SIZE 3

void main()
{
    // Variable definition
    int board[BOARD_SIZE][BOARD_SIZE];
    int player_choice_x;
    int player_choice_y;
    int curr_player = 1;

    // Code Section

    printf("Please enter the cordination where you want to put a sign: ");
    scanf("%d %d", &player_choice_x, &player_choice_y);

    // Check if location is out of range
    if (player_choice_x < 1 || player_choice_x > BOARD_SIZE ||
        player_choice_y < 1 || player_choice_y > BOARD_SIZE)

```

סטנדרטים - דוגמא

• עבודה ע"פ סטנדרטים

• תיעוד הקוד

```
Microsoft Visual C++ - [T:\main.cpp *]
File Edit View Insert Project Build Tools Bsmh Window Help
regis AddChagedBlock

-----
The X - O game
-----
// General : This program simulates the famouse X-O game.
// Input   : A location on the board.
// Process : The location received is checked to be empty or out
//          of range, and if all ok then according to the
//          current player the location is filled with
//          his sign.
// Output  : A message when the location is full or if it is
//          out of range.
-----
// Programmer : Sandy Bar
// Student no  : 056
// Date       : 03.07.1979
-----

Build / Debug / Find in Files 1 / Find in Files 2 / Results / SQL Debugging /
Ready Ln 42, Col 1 REC COL DWR READ
```

חשיבות הסטנדרטים

- הבנת הקוד של התוכניתן לאחר זמן רב.
- הבנת קוד של תוכניתן אחר.
- מניעת טעויות ובאגים ע"י תכנות מבני.
- אחידות במערכת כולה.
- שפה משותפת לתוכניתנים בצוות.
- תיעוד שינויים שנעשו בקוד
- הבנת הרעיון שמאחורי הקוד

סטנדרטים - דברים שאפשר לעשות

- חוברת סטנדרטים...

- שימוש בכלי בקרת תצורה לאיסוף תרגילים שמוגשים

- בניית הציון ממרכיבים בלתי תלויים – סטנדרטים,

- קריאות, בקרת תצורה, עיצוב קוד, ריצה

- שימת דגש חזק על סטנדרטים וקריאות כאשר

- התרגילים עדיין קטנים.

חשיבות העבודה הלמידה העצמית

- אי אפשר ללמד את כל השפות בעולם – חייבים ללמד רעיונות.
- אין אפשרות לתת לתוכניתן חונך צמוד כל הזמן.
- אי אפשר להוציא את התוכניתן לכל קורס, לא תמיד יש קורס מתאים.
- לא תמיד קיים מישהו שמכיר את הנושא (מומחיות).
- בזמן העבודה, על התוכניתן ללמוד נושאים חדשים בכדי לכתוב מודולים מסויימים.

עבודה עצמית – דברים שאפשר לעשות

- בניית "סרגל" ללימוד עצמי
- הכנסת הלימוד העצמי לאורך כל השנים.
- כתיבת תרגילים אשר מסתמכים על לימוד עצמי.
- תרגילים "סגורים" – בכיתות ללא עזרה מחברים.
- מדריך תורן ? האם ניתן להפעיל דבר כזה בבתי הספר.
- מתן ניקוד על עבודה עצמית במבחנים סגורים.
- תרגילי לימוד עצמי.

חשיבות העבודה בצוות

- במערכות גדולות תמיד יהיו מספר תוכניתנים שעובדים במקביל.
- כמו בכל צוות עבודה, גם בצוות תוכנה צריכות לכל אחד להיות מיומנויות אישיות לעבודה בצוות.
- בפיתוח המערכת יש תלויות בין מודולים שונים של תוכניתנים – מודול חבר / מודול נקרא / מודול קורא
- יש משמעות נוספת לבקרת תצורה (Chek-In)
- דורש בדיקות נוספות מהתוכניתנים וסלחנות כלפי חבריהם.

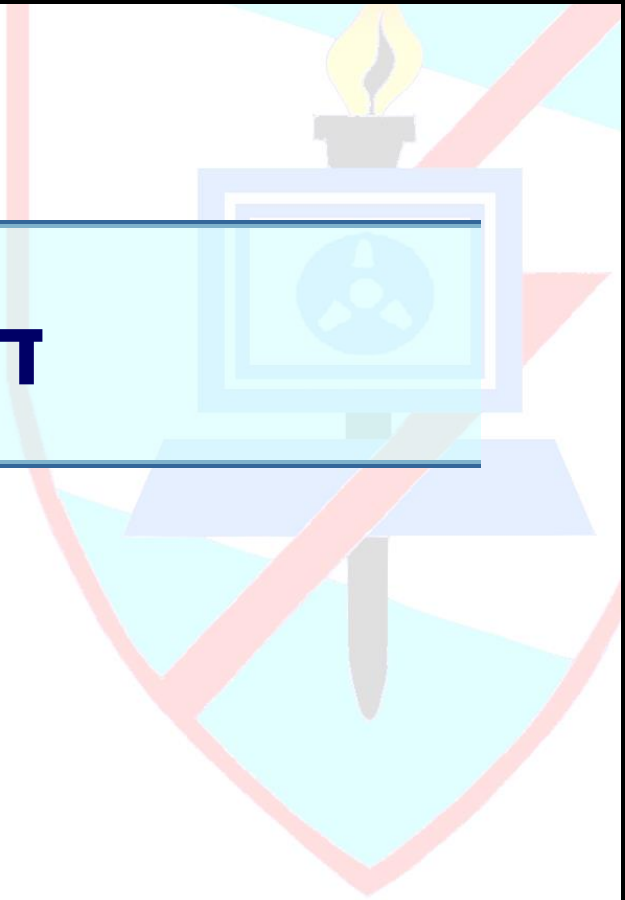
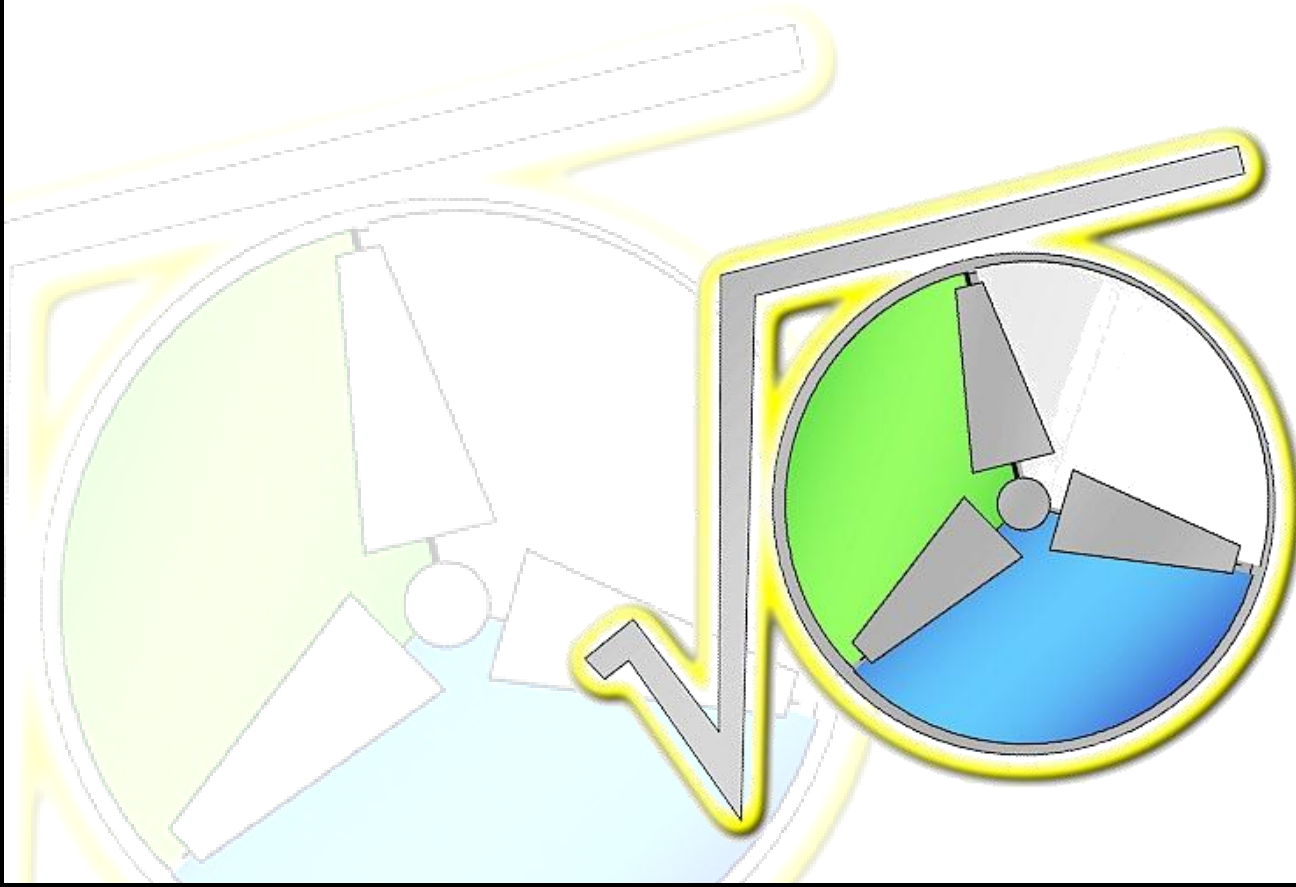
עבודה בצוות – מה עושים

- בניית תרגילים אשר דורשים עבודה בצוות.
- מדרג בעבודת הצוות – הגדלת הצוות בצורה הדרגתית
- בניית תרגילים אשר דואגים לתלויות בין האנשים (כולל תחזוקה של כל של אדם אחר).
- כתיבת תרגיל תחזוקת מערכת הדורש התרגלות לסטנדרטים שונים.

פתרון בעיות – איך מכווינים לפתרון בעיות

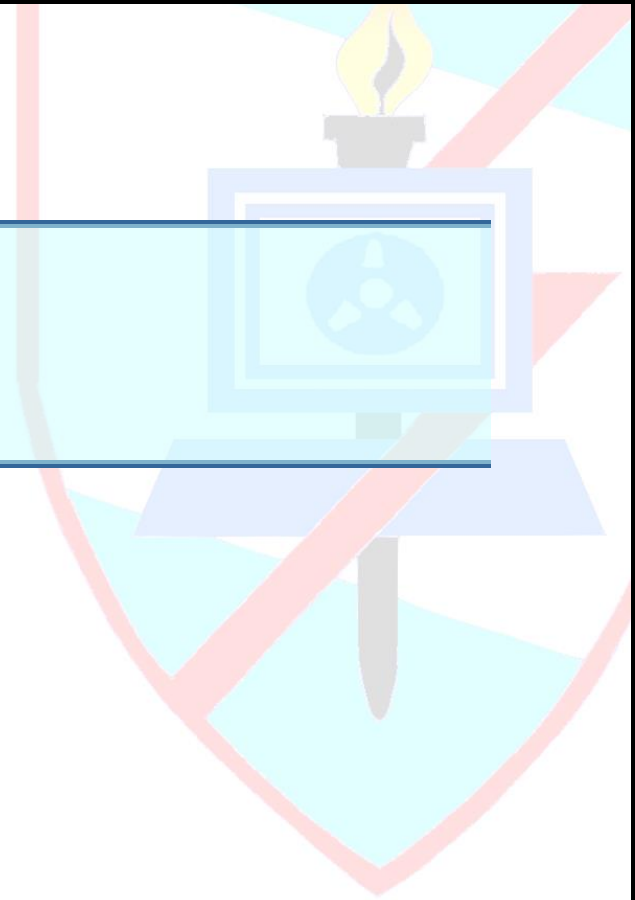
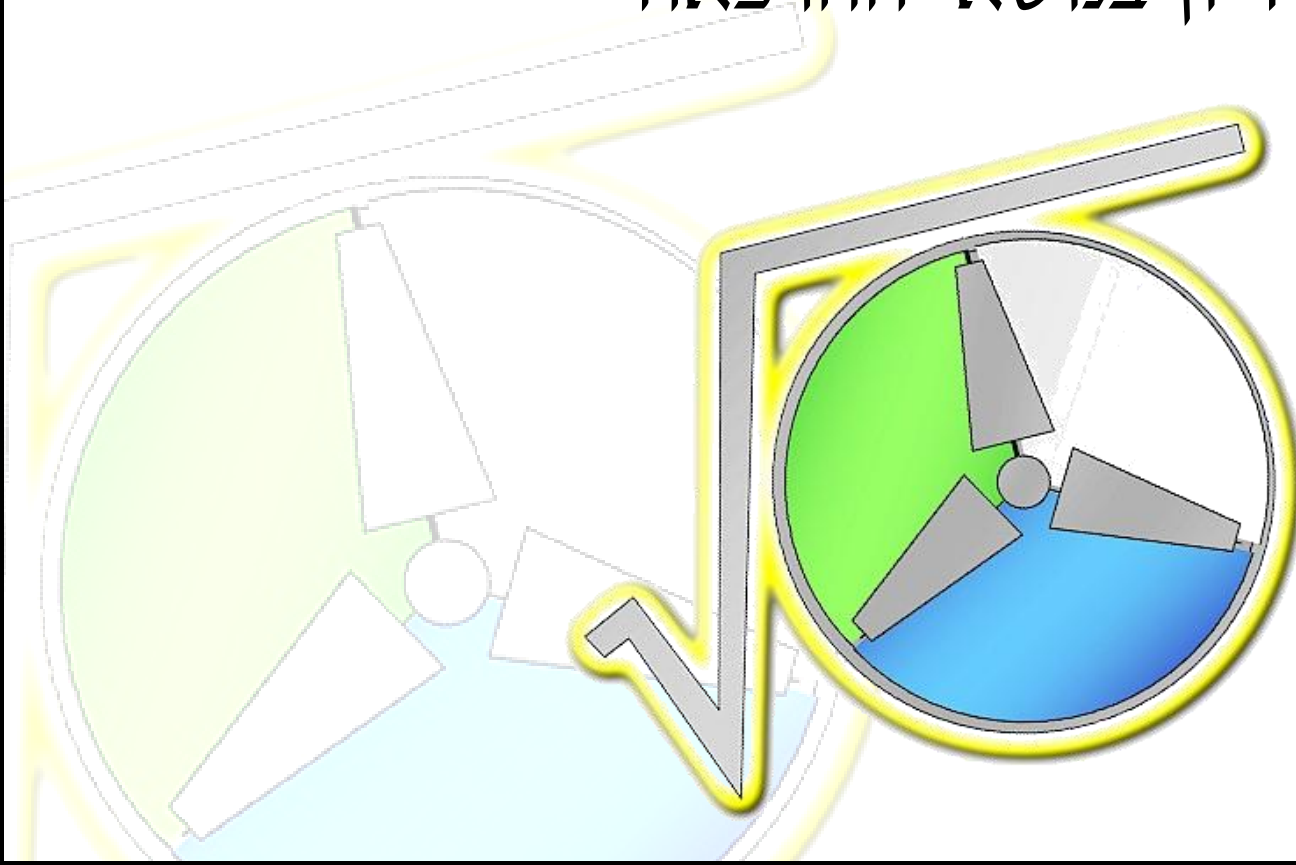
- מספר תרגילים רב. Practice Makes Perfect.
- מתן פתרונות ידועים לכל מיני דברים.
- המחשות צורך לפתרונות.
- חייבים להסביר את המטרה של כל דבר ולא רק את השימוש – מי חשב על הפיתרון? איך הגיעו אליו? מתי הפיתרון הזה טוב?
- יש להיזהר מפורמליות יתר – צריך לחשוב כמו התלמידים ולא לנסות לגרום להם לחשוב כמו מחשב...

דוגמאות נוספות?



שאלות?

דיון בנושאי ההרצאה



תודה!

