

## מדור הבחינות

**אוניברסיטת בר-אילן**

2400123

שנת: תשע"ב סמסטר: 2 מועד: 1 מטלה: 1  
קורס: 02 211 89 תכנות מתקדם 2



## הציון:

### חתימת המרצה:

מס' סידורי \_\_\_\_\_ מתוך \_\_\_\_\_ מחברות

## הוראות לנבחן

1. עליך להבחן בחדר בו הנך רשום.
2. הנח ליד המשיגה בבחינה את חפצריך האישיים כגון: תיקים, ספרים, מחברות, מכשירים סלולריים, קלמרים וכו'.
3. אסור להחזיק בהישג יד חומר הקשור לבחינה/לקורס אלא אם הותר הדבר בכתב על ידי המרצה ורק בהתאם למותר.
4. מסור למשיגה/ה על הבחינה תעודת זהות וכרטיס נבחן חתום ותקף לסמסטר בו מתקיימת הבחינה.
5. **היציאה לשירותים במהלך הבחינה אסורה בהחלט.** נשים בהריון ונבחנים באישור מתאים רשאים לבקש מהמשיגה/ה ולצאת. היציאה בליווי המשיגה/ה ובהתאם לנוהלי האוניברסיטה.
6. נבחן היוצא ללא רשות מחברתו תפסל ותועבר לוועדת משמעת.
7. יש להישמע להוראות המשיגה/ה. אין לעזוב את חדר הבחינה ללא קבלת רשות. חל איסור מוחלט לפנות לנבחנים אחרים בכל עניין ודבר. **בכל עניין פנה למשיגה/ה.**
8. בתחילת הבחינה מלא את פרטיך האישיים ע"ג המחברת. תלמיד שקיבל לידיו שאלון ואין ברצונו

108/302

## ועדת המשמעת מזהירה!

## נבחן שיימצאו ברשותו חומרי עזר

## אסורים או ייתפס בהעתקה,

## ייעגש בחומרה עד כדי

## הרחקתו מהאוניברסיטה.

שנה"ל תשס"ב סמסטר ב' מועד ט'

מס' קורס \_\_\_\_\_ 89-911-02

מחלקה נציג' האחוסס תאריך 15.7.2012

המרצה א"י יריב 66

**מבחן חלק** (אם הבחינה בשני חלקים) \_\_\_\_\_

## הוראות לנבחן בנושא סריקה:

אין לכתוב במחברת בעפרון. יש לכתוב בעט בצבע כחול כהה או שחור בלבד. אין להשתמש בנוזל מחיקה (טיפקס). אין לכתוב בשוליים משני צידי הדף. מחברת בכתב מרושל משפיעה על תוצאות הסריקה.

שם סבל השוליים יחכו ויפלו יקרה. לכן, חל איסור מוחלט



מחזורי המשימה! נכון שימצאו ברשותו חומרי עזר אטורים או יתפס בהעתקה יענש בחומרה עד כדי החרקתו מהארגון בריטון.

תכנות מתקדם 1  
מירי בן-גיסן (קופל)  
התשע"ב / מועד א

כס"ר  
אוניברסיטת בר-אילן  
המחלקה למדעי המחשב



## תכנות מתקדם 2 שאלון סגור

מועד א' תשע"ב

15.7.12

המחלקה למדעי המחשב

89-211-01-02

34	35 נק'	1
15	15 נק'	2
19	20 נק'	3
10	10 נק'	4
18	20 נק'	5
96	100 נק'	סך הכל



48: מתי  
שנת: תשע"ב סמסטר: 2 מועד: 1 מסלול: 1  
קורס: 02 211 89 תכנות מתקדם 2

מרצה: יריב טל.  
מתרגל: איגור רוכלין.  
משך הבחינה: שלוש שעות. אין הארכה.  
חומר עזר: אסור להכניס כל חומר עזר.

הנחיות כלליות: רצוי לענות בגוף הבחינה. אחרי כל שאלה יש מקום לתשובות.  
אם עניתם על שאלה במחברת – ציינו זאת בגוף הבחינה!  
חובה להגיש את המחברת ביחד עם טופס הבחינה בסיום הבחינה.  
חובה לענות על השאלות בעברית, אלא אם יש אישור מהדיקן.

הנחיות טכניות: במידה ונדרשתם לתת נימוק, אזי הוא חובה. כלומר, תשובה לא מנומקת לא תקבל נקודות כלל.  
נימוק לא נכון יגרור פסילת השאלה.  
בכל השאלות – מספר השורות שניתנו לפתרון אינו מרמז על אורך התשובה.

בהצלחה ☺

18  
21  
01  
01  
81  
20



## שאלה 1: Code Improvements (35 נקודות)

נתון הקוד הבא:

```
class CharStringParser
{
public:
    // str is a c-string (terminated with a '\0' character)
    CharStringParser(char *str) : m_str(str) {}

    bool isEmpty() {
        if (m_str == NULL || *m_str == '\0') {
            return true;
        }
        else {
            return false;
        }
    }

    long tryParsePositiveNumericValue() {
        long numericValue;
        if (NULL != m_str) {
            skipLeadingSpaces();

            assert(NULL != m_str);
            for (numericValue=0; isdigit(*m_str); ++m_str) {
                numericValue *= 10;
                numericValue += (*m_str - '0');
            }

            return numericValue;
        }
    }

private:
    // advance m_str until the first non-space character
    void skipLeadingSpaces() {
        assert(NULL != m_str);
        while (isspace(*m_str))
            m_str = m_str + 1;
    }

    char *m_str;
};
```

א. (4 נקודות) האם ניתן להגדיר את `m_str` כטיפוס `const char *` (במקום `char *`), למרות שהפונקציות `skipLeadingSpaces()` ו-`tryParsePositiveNumericValue()` משנות את מיקום המצביע? נמקד!

- כן, ניתן. המשתנה `m_str` הוא `const char *` וכל הפונקציות הנתונות הן `const` ולכן הן לא משנות את המיקום של `m_str` אלא רק את המיקום של המצביע.
- ① `const char * m_str` - המשתנה `m_str` הוא `const char *` ולכן הוא לא משנה את המיקום של `m_str` אלא רק את המיקום של המצביע.
- ② `const char * m_str` - המשתנה `m_str` הוא `const char *` ולכן הוא לא משנה את המיקום של `m_str` אלא רק את המיקום של המצביע.
- המסקנה היא שניתן להגדיר את `m_str` כטיפוס `const char *`.





ב. (4 נקודות) אילו פונקציות יכולות להיות מוגדרות כפונקציות const? נמקו והדגימו כיצד תראה החתימה של הפונקציות לאחר הוספת ה-const.

3/4

הפונקציה `bool isEmpty() const;` תהיה: `const` כי `const` מציין כי הפונקציה לא תשנה את המשתנים שלה.

הפונקציה `long tryParsePositiveNumericValue() const;` תהיה: `const` כי `const` מציין כי הפונקציה לא תשנה את המשתנים שלה.

הפונקציה `void skipLeadingSpaces() const;` תהיה: `const` כי `const` מציין כי הפונקציה לא תשנה את המשתנים שלה.

② `long tryParsePositiveNumericValue() const;`

③ `void skipLeadingSpaces() const;`

(9 נקודות) האם השימוש ב-assert בפונקציה `skipLeadingSpaces()` מוצדק או לא? נמקו. אם השימוש מוצדק הסבירו מדוע בפונקציה `tryParsePositiveNumericValue()` אין `assert` דומה. המז: שימו לב להבדל ברמת ה-`access` (private/public) בין הפונקציות.

ב-`skipLeadingSpaces()` ה-`assert` מוצדק.

הפונקציה `tryParsePositiveNumericValue()` היא private ולכן לא צריך `assert`.

כן, יש להוסיף `assert` כי `m_str` אינו NULL.

הפונקציה `tryParsePositiveNumericValue()` היא public ולכן צריך `assert` כי `m_str` אינו NULL.

כן, יש להוסיף `assert` כי `m_str` אינו NULL.

הפונקציה `tryParsePositiveNumericValue()` היא public ולכן צריך `assert` כי `m_str` אינו NULL.

וכן, יש להוסיף `assert` כי `m_str` אינו NULL.

וכן, יש להוסיף `assert` כי `m_str` אינו NULL.

ד. (18 נקודות) איכות הקוד הנתון שנויה במחלוקת... הציעו לפחות 6 שיפורים אשר עשויים להפוך את הקוד ליותר קריא או להוריד את הפוטנציאל שלו לבאגים. נמקו את הצורך בכל שיפור!

היו ברורים היכן בקוד יש לבצע את השינוי.

שימו לב - הצעה של שינוי משפר את הקוד עלולה לגרום הורדת ניקוד!

הפונקציה `skipLeadingSpaces()` תהיה:

① הפונקציה `skipLeadingSpaces()` תהיה: `void skipLeadingSpaces() { while (m_str != NULL && *m_str == ' ') m_str++; }`

הפונקציה `tryParsePositiveNumericValue()` תהיה:

② הפונקציה `tryParsePositiveNumericValue()` תהיה: `long tryParsePositiveNumericValue() { if (m_str == NULL || *m_str == '\0') return 0; long numericValue = 0; while (m_str != NULL && *m_str == ' ') m_str++; if (*m_str == '+' || *m_str == '-') m_str++; if (*m_str == '\0') return 0; while (*m_str <= '9') { numericValue = numericValue * 10 + (*m_str - '0'); m_str++; } return numericValue; }`

וכן, יש להוסיף `assert` כי `m_str` אינו NULL.

וכן, יש להוסיף `assert` כי `m_str` אינו NULL.

וכן, יש להוסיף `assert` כי `m_str` אינו NULL.

וכן, יש להוסיף `assert` כי `m_str` אינו NULL.

וכן, יש להוסיף `assert` כי `m_str` אינו NULL.

וכן, יש להוסיף `assert` כי `m_str` אינו NULL.

הנחיה: הוסיפו את ה-`assert` כי `m_str` אינו NULL.

4/18

9/1/92  
10/1/92  
Riverside  
Ma  
Kafra  
Owaga  
63  
1/1/93

11/1/92  
12/1/92  
1/1/93

1

1





## שאלה 2: Templates (15 נקודות)

נתון הקוד הבא:

```
template< unsigned long N >
struct Magic
{
    enum { value = (N % 10) + 2 * magic <N / 10>::value };
};

template<
struct Magic<0>
{
    enum { value = 0 };
};
```

א. (5 נקודות) כיתבו אילו אינסטנסיאציות יבצע הקומפיילר ומה יודפס כתוצאה מביצוע שורת הקוד הבאה (נמקו!):  
std::cout << Magic<0>::value << std::endl;

יבוצע שורת הקוד הבאה תבוצע אינסטנסיאציה של specialization עבור 0  
ולכן, Magic<0>::value הוא 0 ולכן תודפס הספרה 0 (נורה שורה)  
✓

ב. (5 נקודות) כיתבו אילו אינסטנסיאציות יבצע הקומפיילר ומה יודפס כתוצאה מביצוע שורת הקוד הבאה (נמקו!):  
std::cout << Magic<1>::value << std::endl;

האינסטנסיאציה שיתבצע: Magic<0> ו Magic<1>  
לכן, Magic<1>::value = 1 + 2 \* Magic<0>::value = 1  
✓

Magic<1>::value = 1 + 2 \* Magic<0>::value

Magic<0>::value = 0

סך הכל יודפס 1 כי 1 + 2 \* 0 = 1 (נורה שורה)  
ג. (5 נקודות) מה מבצע ה-template?  
רמז: העזרו באינסטנסיאציות הבאות בכדי להגיע למסקנתכם:

```
std::cout << Magic<10>::value << std::endl;
std::cout << Magic<11>::value << std::endl;
```

יבוצע Magic<10> תבוצע האינסטנסיאציה הבאה:  
Magic<10>::value = 0 + 2 \* Magic<1>::value  
✓

Magic<10>::value = 0 + 2 \* Magic<1>::value

Magic<1>::value = 1 + 2 \* Magic<0>::value

Magic<0>::value = 0

לכן יודפס 8 (נורה שורה)

הנעם קמחיה





### שאלה 3: Exception Safety (20 נקודות)

נתון הקוד הבא:

```
class Yard {
public:
    Bone *digupBone() {
        Hole *pHole = dig();
        if (pHole && (pHole->hasBone())) {
            Bone *pBone = pHole->getBone();
            releaseHole(pHole);
            return pBone;
        }
        else {
            if (pHole) {
                releaseHole(pHole);
            }
            throw NoBoneException();
        }
    }

private:
    // Allocates a hole.
    // Freeing an unused hole is done by calling releaseHole (do not directly
    // delete the hole! You *must* call releaseHole instead).
    Hole *dig();
    void releaseHole(Hole *pHole) throw();

    std::vector<Hole*> m_holes;
};
```

א. (5 נקודות) נתונות ההנחות הבאות:

- הפונקציות `getBone()`, `hasBone()`, `releaseHole()` מקיימות `nothrow guarantee`.
- שחרור של `Hole` שהוקצה ע"י `dig()` מתבצע ע"י קריאה ל-`releaseHole()` (ולא ל-`delete`).
- הפונקציה `dig()` מקיימת `strong exceptions guarantee`.
- איזה `guarantee` מקיימת הפונקציה `digupBone()`? נמקד!

הפונקציה `digupBone()` מקיימת strong exception guarantee.  
הסיבה היא, כי הפונקציה `releaseHole(pHole)` היא `nothrow` ו-`phole` אינו `NULL`.  
לכן הפונקציה `digupBone()` מקיימת `strong exception guarantee`.

הפונקציה `dig()` מקיימת `strong exception guarantee` והיא `nothrow`.  
לכן הפונקציה `digupBone()` מקיימת `strong exception guarantee`.  
הסיבה היא, כי הפונקציה `releaseHole(pHole)` היא `nothrow` ו-`phole` אינו `NULL`.  
לכן הפונקציה `digupBone()` מקיימת `strong exception guarantee`.

ב. (5 נקודות) איזו בעייה עלולה להיווצר אם הפונקציה `getBone()` (שנקראת מהפונקציה `digupBone()`) יכולה לזרוק `exception`?





המשימה (הפונקציה `getBone()` תזרוק `exception` אם `id` אינו `releaseHole(pHole);` אם יש חור אחד בלבד.   
 בלוח, חסוייה הפונקציה `digupBone()`.

ג. (10 נקודות) כל `Hole` שמוחזר ע"י `dig()` יש לשחרר כשמסיימים להשתמש בו ע"י קריאה ל-  
`releaseHole()`.

בתוכנית הנתונה שחרור זה מתבצע ידנית (ישנן קריאות ל-`releaseHole()` במקומות המתאימים).  
 כתבו `Guard` אשר יהיה אחראי לשחרור ה-`Hole` במקום הקריאות המפורשות ל-`releaseHole()`. הדגימו  
 היכן וכיצד יש לבצע את איתחול ה-`Guard` בפונקציית `digupBone()`.

9/10

`class HoleGuard {`

`public:`

`HoleGuard() {`

`m_pHole = dig();`

`}`

`~HoleGuard() {`

`if (pHole) {`

`releaseHole();`

`}`

`}`

`private:`

`Hole* m_pHole;`

`}` `digupBone` הפונקציה `digupBone` יקראת במקום

המורה `Hole* pHole = dig();` וזו אוליית `HoleGuard` מוגדרת

אז: `HoleGuard hGuard = new HoleGuard();`

אם נצטרך לעצור את הקריאה המפורשת ל-`releaseHole()` והקריאה תתבצע  
 כאשר תאבד הבסיסית.

++C : אוליית `new` עם `stack` לא מאתחלים

die

2

✓  
ttd : changed, to a-also 3d instance  
or was.



## שאלה 4: Java (10 נקודות)

א. הסבירו בקצרה: מה זה Servlet, מה זה JSP ומה ההבדלים ביניהם.

Servlet ו JSP לאנשים חתום לשרת web ב Java

ההבדלים בין Servlet ל JSP:

1) Servlet הוא קוד Java שכתוב ממשק HTML (נשלט באמצעות הקוד הקודי)

הקוד ו JSP הוא קוד HTML שניתן לשרת בו קוד Java. (נשלט באמצעות הקוד הקודי HTML)

2) JSP מתקבל ב Runtime והוא נכתב ב Servlet יש קודים ל JSP קודים.

הפכה Servlet:

Servlet מקבל ממשק request ומחזיר response.

מאפשר תמיכה בין ה post והן ה get (ב get המעצבים משתמשים ב URL, יש מקום מוגדר ובעזרתו חתום למערכת או post מא).

המשק כמחזורי

ב. הסברו בקצרה מה זה Reflection ומה השימוש במנגנון הזה.

Reflection הוא מנגנון אשר מספק מידע

באמצעות Reflection נוכל לקבל מידע ב Runtime על מחזורי: על הקונסטנטות

(constructors) ואם, על המתודות (Methods) ואם, על המשתנים (Fields) ואם ואם...

ניתן לקבל מידע זה אם אדם יודע לקבל

בנוסף, נוכל לקבל מידע על המתודות בדרך בינאית, על invoke.

## שאלה 5: Java (20 נקודות)

א. (5 נקודות)

הסבירו בקצרה למה נועד תרשים המחלקות (class diagram)?

למה נועד תרשים זרימה (flow diagram)?

תרשים המחלקות ותוספת זרימה הם חלק מ UML





בנימין ונתן לו מלכות

ଏହାକୁ flow diagram ବୋଲି କୁହାଯାଏ।

Ng"bN

ବିଜ୍ଞାନୀମାନେ ମହାକାଶ ଚାଷିମାନଙ୍କ ସହ ମିଶି, ମାଧ୍ୟମିକ

ביאכאמא 15 עס זיך מ'האט ס'מינים פונעם העלפנים דער ערשטער איז זיך

ב. מחלקת התקצוב של חברה "X" מעוניינת להקים מערכת מידע אוטומטית ככל הניתן שתתמוך בהפקת דוחות שכר וחישוב רווח או הפסד שנתי.

המערכת מקבלת מידע אודות עובדי החברה (מס' זהות, שם תפקיד, תאריך התחלת עבודה) ומעבדת אותו באופן הבא: בתחילת כל חודש, המערכת מפיקה דו"ח משכורות העובדים. המשכורות מחושבות על סמך טבלת תפקידים (המתאימה לכל תפקיד סכום שקלי) וטבלת ותק (המתאימה לתקופת ותק מסוימת מקדם הכפלה, לדוגמא: עובד עם ותק בין 3 ל 5 שנים, משכורתו תוכפל ב- 1.2).

בנוסף מערכת המידע מנהלת את תקציב החברה.

לכל תקציב יש: מספר תקציב, תקופת תקציב, סכום התחלתי ואחראי או אחראים לתקציב, שהם עובדי החברה. כל עובד יכול להיות אחראי למספר תקציבים. אחראי תקציב יכול להגיש בקשות להוצאות סכומים מתקציבו. כל בקשה תכיל את סיבת הבקשה, תאריך והסכום המבוקש.

- (4 נקודות) מיהם ישויות (actors) של המערכת?

היתשוקה מחזק

ה'מ"ו (vectors) ו' המרחב הם המרחב המ"ו ו' המרחב

[illegible]

~~החשבון החדש הוצא על ידי מנהל המכס, וכן, ה' מנהל המכס הוצא על ידי מנהל המכס, ובקרה.~~

~~החומר, במקרה זה ה- vectors הם 'המשוואות' המתוארות ה-  $u, v, w$  הן 'ההתאמות'.~~

১৯৯০ খ্রি: ৩

- (4 נקודות) מהם הטרונוקציות (use cases) העיקריות של המערכת?

[illegible]

ଅବସ୍ଥା — ଶୁଦ୍ଧତା : ଶୁଦ୍ଧତା ②

המשק החלמיש הקטן

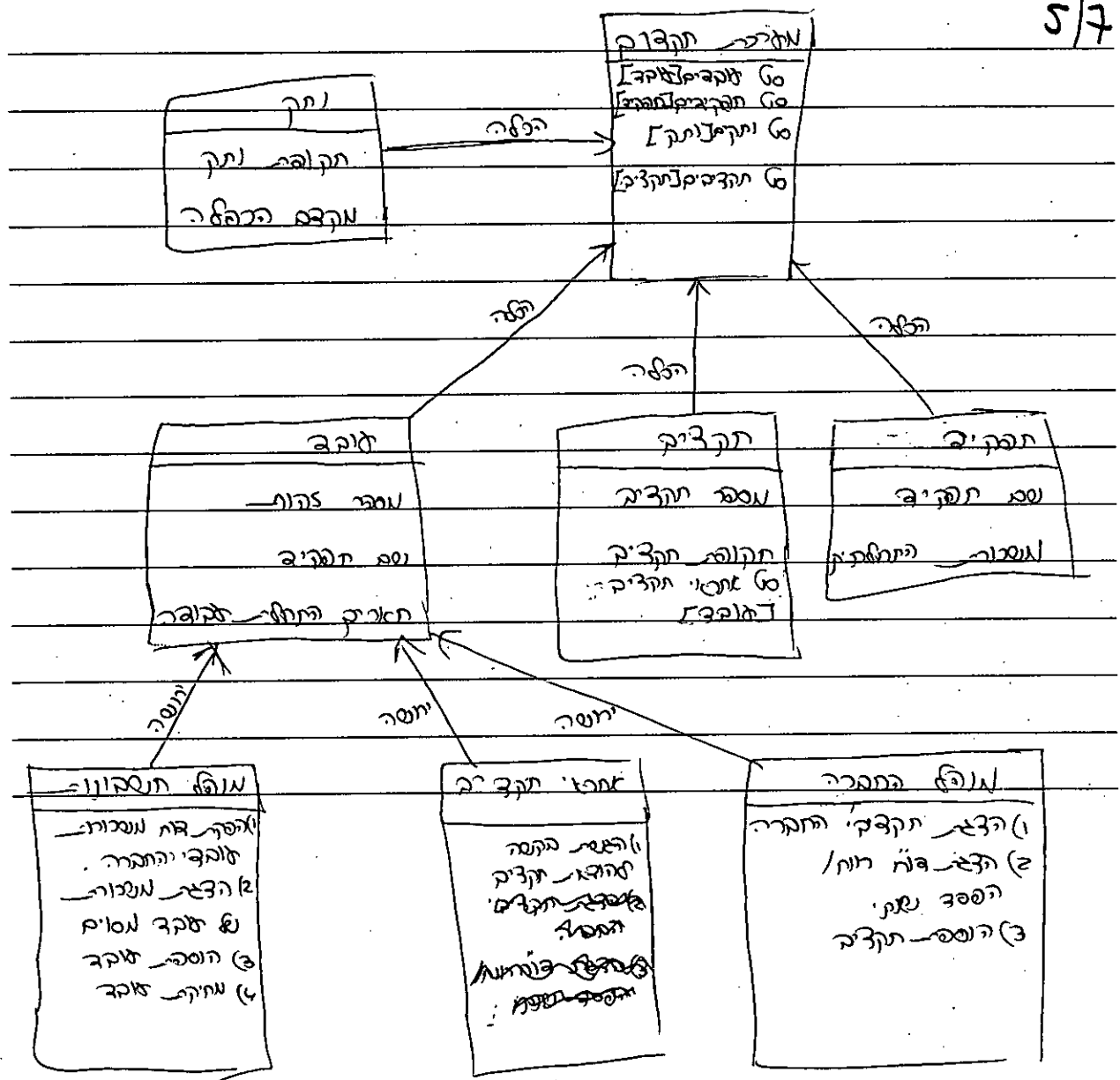


**בד**

⑦ הוספת עובד	③ הוצאת תקציב בחברה
⑧ הוספת תקציב	④ הוצאת צ"ח חות' / הפסד שוק
	⑤ הוצאת משכורת ולא עובד מסוים
	⑥ מחיקת עובד

(7 נקודות) צרו תרשים המחלקות של המערכת (class diagram). עבור כל מחלקה, רשום את תכונותיה והקשרים המבניים למחלקות אחרות.

5/7



— תסר: 'אם אין' תקציב לאחראי תקציב.  
— 'אם אין' עובד לעבוד/אין.  
— 'אם' ההכנסה: (1,1), (1,1), (1,1)





706

$$\text{value} = 1 + 2 \cdot \text{magic} \langle 32 \rangle :: \text{value}$$

4321

$$\text{magic} \langle 32 \rangle :: \text{value} = 2 + 2 \cdot \text{magic} \langle 3 \rangle$$

$$\text{magic} \langle 3 \rangle :: \text{value} = 3 + 2 \cdot \text{magic} \langle 4 \rangle$$

$$\text{magic} \langle 2 + 2 \cdot 3 \rangle = 12$$

$$1 + 2^0 + 2^1 + 2^2$$

$$1 + 4 + 12$$

$$1 + 2(2 + 2(3 + 2 \cdot 4))$$

$$1 + 2(2 + 2 \cdot 2) = 12$$

$$1 + 4 + 12 + 4 \cdot 8$$

$$1 + 4 + 12 + 32 =$$

$$0 + 2 \cdot \text{magic} \langle 10 \rangle :: \text{value} = 4 \quad : 100$$

$$\text{magic} \langle 10 \rangle$$

$$\text{value} = 0 + 2 \cdot \text{magic} \langle 1 \rangle = 2$$

$$\text{magic} \langle 1 \rangle = 1 - 0 = 1$$

הערות המרצה

שאלה 4

המשק מס' 4

הסקר JSP : (Java Server Pages)

למשל יצרנו מסמך בצורה נוחה והדפסנו את HTML וכן

ע"י out.println

שאלה 5

מס' 5

היטוי של המערכת - תם :

1. מערכת - תקציר ✓

~~2. מערכת - מערכת יצרנית~~  
~~3. מערכת - מערכת יצרנית~~  
~~4. מערכת - מערכת יצרנית~~  
~~5. מערכת - מערכת יצרנית~~

2. מערכת - תקציר

3. מערכת - מערכת

4. מערכת - מערכת

5. מערכת

1-24 4/14 11.4.1988 11.4.1988

1



המחלקה לבריאות הציבור

הערות המרצה

## הערות המרצה



## הערות המרצה

04571061 11000 215027 287 281

## הערות המרצה

הערות המרצה

הערות המרצה



הערות למרצה

---

הערות המרצה





