

המחלקה להנדסת מערכות מידע

25.08.08

אוניברסיטת בן-גוריון בנגב

מרצה: ארנון שטורם
מתרגל: איגור דבורקין

מבחן מועד א'

ניתוח ועיצוב מונחה עצמים – 372-1-3103
סמסטר אביב, תשס"ח

פתרון חלקי ואפשרי

הוראות כלליות

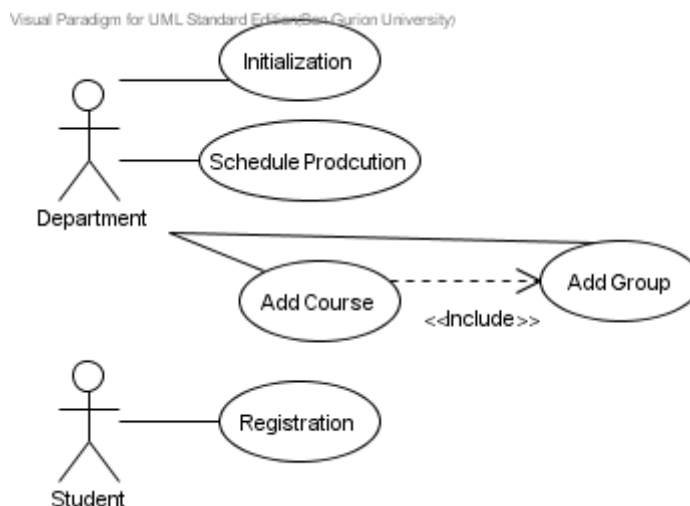
- ☐ משך המבחן: שעתיים וחצי שעות.
- ☐ מבחן זה מכיל 10 דפים, כולל דף זה.
- ☐ במבחן 3 שאלות.
- ☐ יש לענות על כל השאלות על שאלון המבחן בלבד.
- ☐ מותר השימוש בכל חומר עזר, פרט לאמצעים ממוחשבים כגון מחשב נייד וטלפון סלולרי.
- ☐ יש להקפיד על כתיבה ברורה ומסודרת של התשובות.
- ☐ אם הנכם מוצאים צורך להניח הנחות כלשהן, ציינו אותן במפורש ונמקו.

בהצלחה!!

שאלה 1 – Structural + Dynamic Modeling – 50 נקודות

לפניכם תיאור מערכת של ניהול קורסים באוניברסיטה. מערכת זו מאפשרת הגדרת קורסים וקבוצות וכן המרצים המלמדים קבוצות אלו, כמו גם רישום סטודנטים לקורסים אלו. לכל קורס מוגדרות הנק"ז שלו; לכל קבוצה מוגדרים מספר המקומות המקסימלי, השעות והמיקום. מערכת השעות היא שנתית. ניהול הקורסים והרישום אליהם נעשה במסגרת מחלקתית. כלומר, לכל מחלקה קיימת רשימת הקורסים, הקבוצות והמרצים ולכל מחלקה יש את מערכת השעות שלה. המערכת תומכת בפונקציונליות הבאה: אתחול, הוספת קורס, הוספת קבוצה, בניית לוח שעות למחלקה, ורישום סטודנטים לקבוצות. כל סטודנט יכול להירשם לקבוצה רק פעם אחת.

א. (5 נקודות) בנו את תרשימי נסיבות השימוש של המערכת.



ב. (7 נקודות) תארו את נסיבת השימוש של רישום סטודנט לקורסים. תאור נסיבת השימוש צריך להיות מקיף ולכלול: את השחקנים, תנאים מקדימים, תנאים סופיים, תרחיש עיקרי, פעילויות נוספות אפשריות ותרחישי כשל.

Use Case: Registration

Actors: Student

Purpose: Record student registration to courses and sections.

Preconditions: Schedule exists
 Student is known to the system

Postconditions: Student is registered to courses and sections

Main flow:

1. Student which to register.
2. The student provides his identifier.
3. The system presents the courses offered by the department.
4. The student selects a course.
5. The system presents the sections available for that course.
6. The student selects the desired course.
7. The student selects a section.
8. The student registers to that section.
9. The system stores the registration.
- In case other registration is required return to 4.
10. The student finalizes the registration.
11. The system producer a registration report.

Alternative flow:

8. No available place in the selected section.
 The system indicates an error.
 Return to 7.
10. Student would like to cancel one of its registered section.
 A new flow should be indicated here.

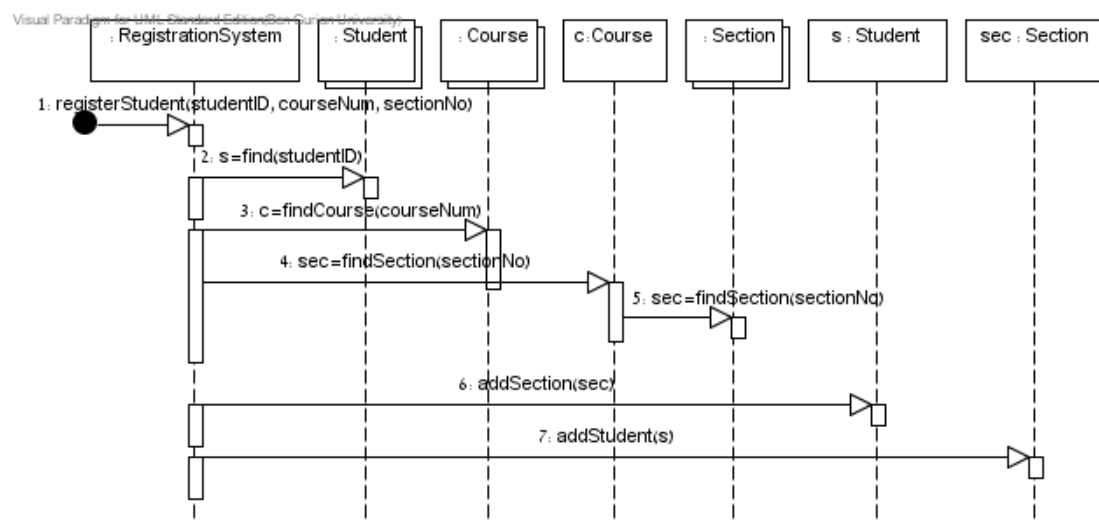
Failure flows:

At any stage the system crash. The system should be rebooted.

ג. (8 נקודות) כתבו את החוזה של פעולת רישום סטודנט לקבוצת קורס.

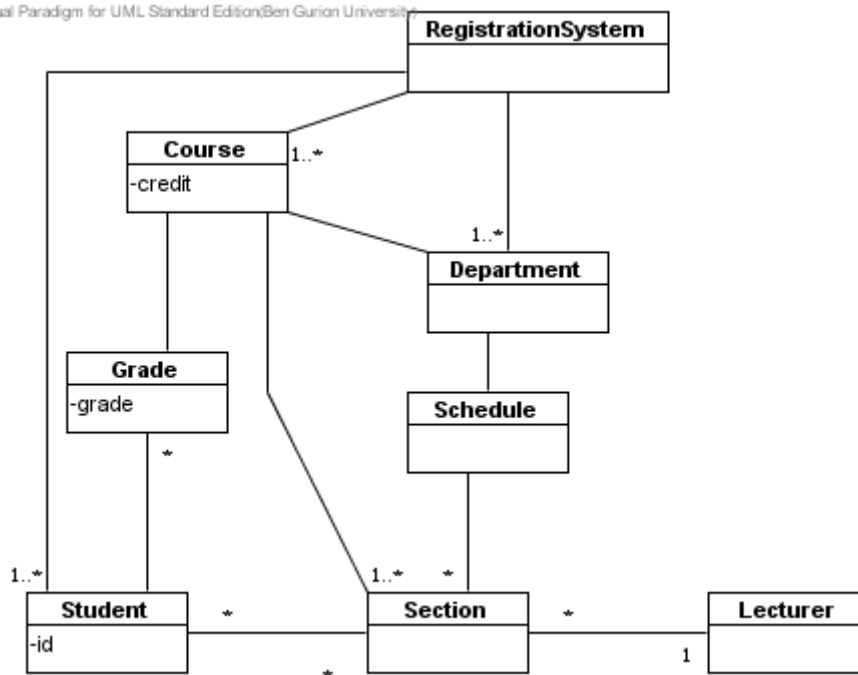
Contract Name: registerStudent(studentID, courseNum, sectionNo)
 Reference: Use Case: Registration
 Precondition: Student, Course, and Section are known to the system.
 Postcondition: An association is created between the student and the section.
 An association is created between the section and the student.

ד. (5 נקודות) בנו את תרשימי הרצף המקיים את תנאי הסיום של פעולת רישום סטודנט לקבוצת קורס.



ה. (15 נקודות) בנו תרשים מחלקות של שכבת התחום.

Visual Paradigm for UML Standard Edition (Ben Gurion University)



1. (10 נקודות) לפניכם מספר אילוצים בשפה טבעית למערכת שתוארה לעיל. עליכם לבדוק האם האילוץ נדרש. באם הוא נדרש יש לכותבו ב-OCL, אחרת יש להסביר מדוע אינו נדרש.

1. לכל קבוצה מרצה אחד.

לא נדרש – מופיע בתרשים המחלקות.

2. מערכת השעות של מחלקה יכולה לכלול רק קבוצות של קורסים השייכים אליה.

Context: Department

Inv: course->include (schedule.section.course)

3. המזהה של סטודנט הוא חד ערכי.

Context: RegistrationSystem

Inv: students->forAll(s1,s2| s1.id= s2.id implies s1=s2)

4. סטודנט יכול להירשם לקורסים עד מקסימום של 20 נק"ז.

Context: Student

Inv: section.course.credit->sum()<=20

5. סטודנט אינו יכול להירשם לקורס אותו הוא עבר (ציון מעל 55).

Context: Student

Inv: section.course->forAll(c1, c2| c1=c2 implies (grade.grade <55 && (grade.course =c1 || grade.course=c2)))

שאלה 2 – State Charts – 25 נקודות

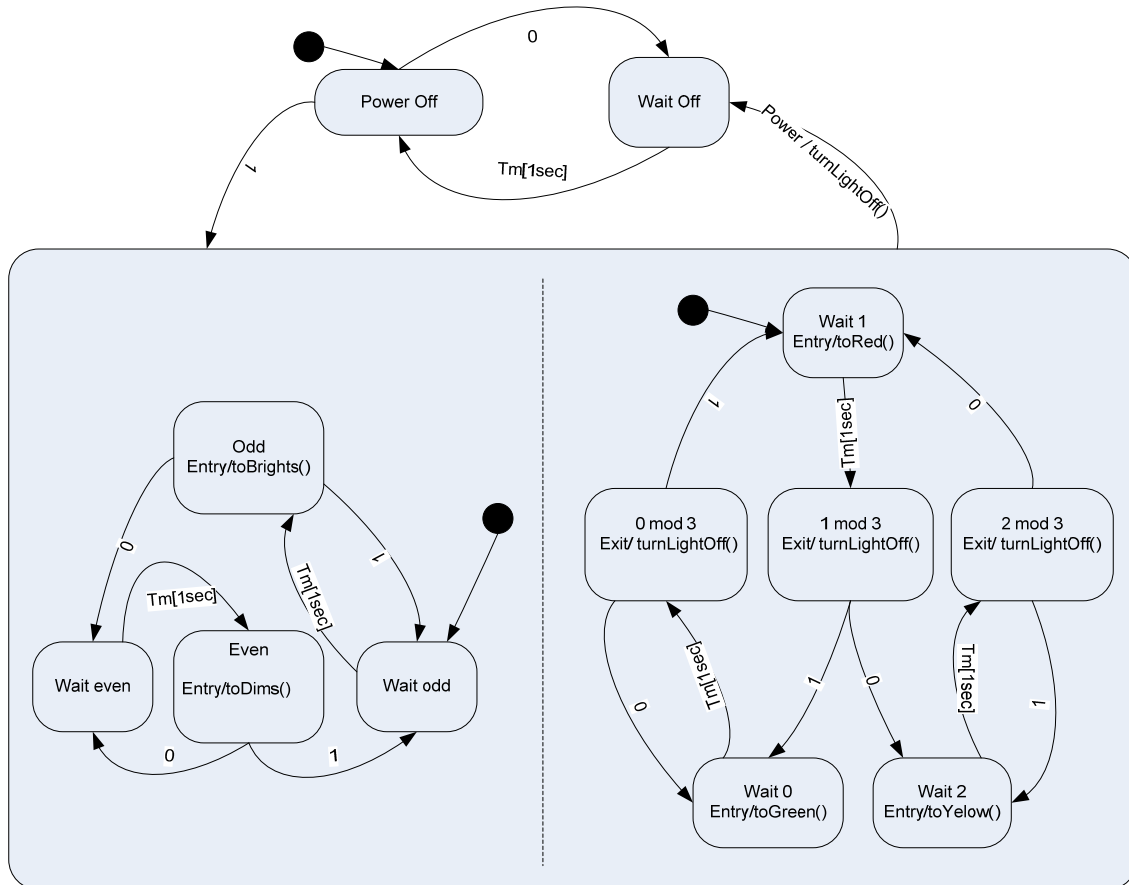
המערכת "רמזור מטורף" מכילה שלושה כפתורים: power, 1 ו-0. בנוסף לכך המערכת מכילה שלוש נורות בצבעים: ירוק, צהוב ואדום. כאשר כל נורה יכולה להיות כבויה, דולקת בעוצמה חלשה או דולקת בעוצמה חזקה. לאחר כל לחיצה על כפתור, המערכת מבצעת השהייה של שנייה ורק אז מאפשרת לבצע לחיצה נוספת.

המערכת נדלקת ע"י לחיצה על 1, ברגע זה המערכת מאתחלת משתנה מטיפוס מחרוזת שערכו "1". החל מרגע זה, כל לחיצה על הכפתורים 1 ו-0 מעדכנת את המחרוזת ע"י שרשור תו תואם לשם הכפתור. לדוגמה: לאחר לחיצה על $1 \leftarrow 0 \leftarrow 0 \dots$ המחרוזת תתעדכן ל "100...".

עוצמת התאורה משתנה שנייה אחת אחרי לחיצה על כפתורי המספרים, כאשר ערך משתנה המחרוזת הוא מספר זוגי לפי בסיס עשרוני התאורה חלשה ואילו כאשר הערך אי-זוגי לפי בסיס עשרוני התאורה חזקה. צבע הנורה הדולקת (בכל רגע דולקת רק נורה אחת) הוא ירוק כאשר שארית החלוקה של ערך המחרוזת לפי בסיס בינארי מתחלק ב-3 ללא שארית, אדום כאשר מתחלק ב-3 עם שארית 1 ואילו צהוב כאשר מתחלק ב-3 עם שארית 2. צבע הנורה משתנה ברגע הלחיצה על כפתורי המספרים.

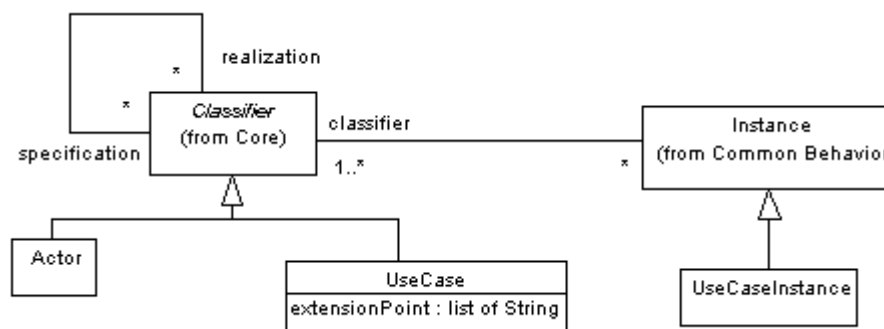
המערכת נכבית ע"י לחיצה על power, ברגע זה המונה מתאפס. במצב כבוי, לחיצה על 0 לא משנה את מצב המערכת, ואילו לחיצה על 1 מאתחלת אותה מחדש.

עליכם לבנות מכונת מצבים אשר מתארת את המערכת הנ"ל.



שאלה 3 – Metamodeling – 25 נקודות

לפניכם מודל-על חלקי של UML2



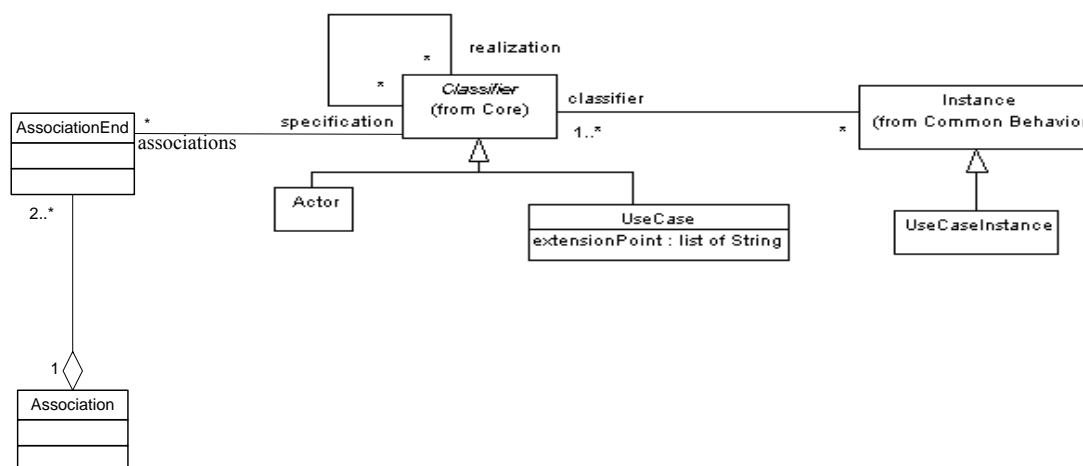
1. (5 נקודות) הגדירו את קשר האסוציאציה (Association), ותארו את מאפייניו.

An association relationship between classifiers indicates some meaningful and interesting connection. In UML, associations are defined as "the semantic relationship between two or more classifiers that involve connections among their instances."

Associations also represent binding between actor and use case in UCD. The arrow indicates that the actor has an interaction between the actor and the use case. This association is binary.

Association is characterized by its ends (including multiplicity and navigational properties).

2. (10 נקודות) האם קשר אסוציאציה זה מתואר בתרשים? אם כן, תארו באיזה אופן, אחרת הרחיבו את מודל העל כדי שיתמוך בקשר זה.



3. (4 נקודות) הגדירו את האילוי: שחקן (Actor) יכול להיות בקשר אסוציאציה רק עם תרחישי נסיבות (UseCase) וקשרים אלה הם בינאריים.

Context Actor inv:

```
self.association.Associations->forAll(a | a.connection->size = 2 and
a.allConnections->exists(r | r.type.ocIsKindOf(Actor)) and
a.allConnections->exists(r | r.type.ocIsKindOf(UseCase)))
```

4. (6 נקודות) הגדירו לפחות 3 אילוצים משמעותיים נוספים, רשמו כל אחד מהם גם בשפה טבעית וגם בשפת אילוצים פורמאלית (OCL). במידת הצורך ניתן להרחיב את המודל.

For each **Operation** in an offered **Interface** the **Actor** must have a matching **Operation**.

Context:Actor

```
Inv: self.specification.allOperations->forAll (interOp |
self.allOperations->exists ( op | op.hasSameSignature (interOp) ) )
```

UseCases can only have binary **Associations**.

Context:UseCase

```
Inv: self.associations->forAll(a | a.connection->size = 2)
```

UseCases can not have **Associations** to **UseCases** specifying the same entity.

Context:UseCase

```
Inv: self.associations->forAll(a | a.allConnections->forAll(s, o|
s.type.specificationPath->isEmpty and o.type.specificationPath->isEmpty
or
(not s.type.specificationPath->includesAll(o.type.specificationPath) and
not o.type.specificationPath->includesAll(s.type.specificationPath)) ) )
```

A **UseCase** can only have 'uses' or 'extends' **Generalizations**.

Context:UseCase

```
Inv: self.generalization->forAll(g | g.stereotype.name = 'Uses' or g.stereotype.name =
'Extends')
```

A **UseCase** cannot contain any **Classifiers**.

Context:UseCase

```
Inv: self.contents->isEmpty
```

For each **Operation** in an offered **Interface** the **UseCase** must have a matching **Operation**.

Context:UseCase

```
Inv: self.specification.allOperations->forAll (interOp |
self.allOperations->exists ( op | op.hasSameSignature (interOp) ) )
```