

אוניברסיטת בן-גוריון

מדור בחינות

מספר נבחן: _____

בגיליון הבחינה 6 עמודים, אנא בדוק כי גיליון הבחינה שלם. לבחינה מצורף גיליון תשובות. רשום/רשמי תשובתיך בגיליון התשובות בלבד – תשובות מחוץ לגיליון לא יבדקו.

בהצלחה!

תאריך הבחינה: 7.2.2003

שם המורה: ד"ר מיכאל אלחדד, מר אלעד שילר, מר מנחם אדלר

שם הקורס: תכנות מערכות

מספר הקורס: 202-1-2031

מיועד לתלמידי: מדעי המחשב, הנדסת תוכנה

שנה: תשס"ג סמסטר: א' מועד: א'

משך הבחינה: שלוש שעות חומר עזר: אסור

(20 נקודות)

שאלה 1

סידרת פיבונאצ'י מוגדרת על ידי:

$$\begin{aligned}a_0 &= 0 \\ a_1 &= 1 \\ a_n &= a_{n-2} + a_{n-1}\end{aligned}$$

המחלקה הבאה מייצגת מספר בסדרת פיבונאצ'י:

```
class Fib {  
  
    private int prev;  
    private int curr;  
  
    Fib() {  
        prev = 0;  
        curr = 1;  
    }  
  
    public int synchronized getValue() {  
        return prev;  
    }  
  
    public synchronized Fib next() {  
        int tmp = prev;  
        prev = curr;  
        curr += tmp;  
  
        return this;  
    }  
}
```

א. הוסף למחלקה מתודת בדיקה בשם test המחזירה ערך אמת אם מתקיימת התכונה הנשמרת (Invariant) של מצב המחלקה: $curr \geq prev$ הם שני מספרים עוקבים בסדרת פיבונאצ'י. (10 נקודות)
[חשוב/חשבי כיצד ניתן לשחזר ולבדוק את כל אברי הסדרה, משני האברים האחרונים (הנתונים כ- $curr$ ו- $prev$) ועד לאיבר הראשון - 0]

ב. שכתב את המחלקה המקורית, כך ש- next לא מסונכרנת, ובכל זאת המחלקה בטוחה. (10 נקודות)

שאלה 2 (10 נקודות)

שאלה 2

נתונה הגדרה של מחלקה Summer בשפת Java:

```
public Class Summer {  
  
    private int sum;    // Automatically initialized to zero.  
  
    public synchronized void up() {  
        sum++;  
        System.out.print(sum);  
    }  
    public synchronized void down() {  
        sum--;  
        System.out.print(sum);  
    }  
}
```

בניה שתהליך מסוים מורכב משישה thread-ים אשר ניגשים לאובייקט אחד מסוג Summer ששמו exam, כאשר שלושה מהם מפעילים את exam.up() ושלושת האחרים מפעילים את exam.down(). הפלט של התהליך הזה יהיה מורכב משישה מספרים בני ספרה אחת. למשל, הפלט 121010 יתקבל כתוצאה מביצוע ההפעלות בסדר הבא:

```
exam.up();  
exam.up();  
exam.down();  
exam.down();  
exam.up();  
exam.down();
```

כמובן גם פלט דוגמת 1010-10 אפשרי.

שנה/י את התכנית הנ"ל כך שיהיה מובטח שאף אחד מהמספרים המודפסים לא יהיה שלילי. כלומר הפלט 10-1010 יהיה בלתי אפשרי.

שאלה 3 (10 נקודות)

שאלה 3

נתון הקוד הבא ב-C++:

```
class B {  
public:  
    B(int i){ m_i = i;}  
private:  
    int m_i;  
};  
  
class A {  
public:  
    A(){}  
private:  
    B b;  
};  
  
int main() {  
    A a;  
    return 0;  
}
```

- מדוע הקוד הנ"ל לא מתקמפל? (5 נקודות)
- תקן/ני את הקוד כך שיעבור קומפילציה. (5 נקודות)

המחלקות A ו B מוגדרות בשפה Java באופן הבא:

```
class A {
    protected int x_;
    private A a_;

    public A(int x) { x_ = x; }

    public void setA(A a) { a_ = a; }

    public int sum() {
        int s = x_;
        if (null == a_) {
            return s;
        } else {
            return s + a_.sum();
        }
    }

    public int doubled() { return x_ + x_; }

    public static void main(String args[]) {
        System.out.println("Test");

        A a1 = new A(1);
        System.out.println("a1.sum = " + a1.sum());
        A a2 = new A(2);
        a1.setA(a2);
        System.out.println("a1.sum = " + a1.sum());

        A a3 = new B(2, 3);
        System.out.println("double a3 = " + a3.doubled());
    }
}

class B extends A {
    protected int y_;
    public B(int x, int y) { super(x); y_ = y; }

    public int doubled() { return 2 * (x_ + y_); }
}
```

- א. מה יהיה הפלט של הרצת הפונקציה main במחלקה A? (5 נקודות)
- ב. הגדר/י את המחלקות A, B ואת הפונקציה main בשפת C באופן שקול להגדרתן ב Java על ידי השלמת הקוד החסר בתבנית. (15 נקודות)

בתונה מחלקה IncNumber המממשת את הממשק Incractable באופן הבא:

```
public interface Incractable
    extends java.io.Serializable
{
    public void increase();
}

class IncNumber
    implements Incractable
{
    protected long num_;
    protected long offset_;

    public IncNumber(long n, long o) {
        num_ = n;
        offset_ = o;
    }

    public synchronized void increase() {
        num_ += offset_;
    }

    public long getNum() { return num_; }
}
```

בתונה מחלקה RemoteIncNumber המממשת את הממשק RemoteIncractable באופן הבא:

```
public interface RemoteIncractable
    extends java.rmi.Remote
{
    public void increase()
        throws java.rmi.RemoteException;
}

public class RemoteIncNumber
    extends java.rmi.server.UnicastRemoteObject
    implements RemoteIncractable
{
    protected long num_;
    protected long offset_;

    public RemoteIncNumber(long n, long o)
        throws java.rmi.RemoteException {
        num_ = n;
        offset_ = o;
    }

    public synchronized void increase()
        throws java.rmi.RemoteException {
        num_ += offset_;
    }

    public long getNum() { return num_; }
}
```

במחשב H1 מוגדר משתנה בשם progressor1 מטיפוס ProgressorImpl המממש מממשק Progressor באופן הבא (פונקצית progress מקדמת את element שהיא מקבלת ב n צעדים):

```
public interface Progressor
    extends java.rmi.Remote
{
    public void progress(Increasable element, int n)
        throws java.rmi.RemoteException;
    public void progress(RemoteIncreasable element, int n)
        throws java.rmi.RemoteException;
}

public class ProgressorImpl
    extends java.rmi.server.UnicastRemoteObject
    implements Progressor
{
    public ProgressorImpl()
        throws java.rmi.RemoteException { }

    public void progress(Increasable element, int n)
        throws java.rmi.RemoteException {
        for (int i=0; i<n; i++)
            element.increase();
    }

    public void progress(RemoteIncreasable element, int n)
        throws java.rmi.RemoteException {
        for (int i=0; i<n; i++)
            element.increase();
    }
}
```

המשתנה progressor2 במחשב H2 הוא reference למשתנה progressor1 של H1.

א. תאר/י את הרצת שורות הקוד הבאות ב H2:

```
IncNumber element = new IncNumber(2,7);
progressor2.progress(element, 3);
System.out.println(element.getNum());
```

ב. תאר/י את הרצת שורות הקוד הבאות ב H2:

```
RemoteIncNumber remoteElement = new RemoteIncNumber(2, 7);
progressor2.progress(remoteElement, 3);
System.out.println(remoteElement.getNum());
```

התיאור בסעיפים א-ב יתבסס על הסכמה בגיליון התשובות באופן הבא:

- קריאה מ-H1 ל-H2 (ולחפך) תצוין על ידי חץ מ-H1 ל-H2 (ולחפך) ומעליו שם האובייקט עליו מבוצעת הפעולה, שם הפונקציה הנקראת, והפרמטרים הנשלחים לפונקציה.
- החזרת ערך מ-H2 ל-H1 (ולחפך) תצוין על ידי חץ מ-H2 ל-H1 (ולחפך) ומעליו הערך המוחזר.
- ציין (בריבוע מתחת H2 בתרשים) את הערך המודפס על ידי H2.

שאלה 6

(16 נקודות)

נתונים 4 מודלים של שרת-לקוח (Client-Server):

- מודל 1: שרת המנהל קשר TCP (TCP connection) עם לקוח אחד בכל זמן על ידי ת'רד (thread) אחד.
 מודל 2: שרת המנהל מספר קשרי TCP (TCP connection) עם מספר לקוחות על ידי מספר ת'רדים - ת'רד עבור כל קשר עם לקוח.
 מודל 3: שרת המנהל מספר קשרי TCP (TCP connection) עם מספר לקוחות על פי תבנית ה-Reactor - מאגר ת'רדים (thread pool) עם Selector מעל גישה ל I/O ב non-blocking mode.
 מודל 4: שרת המנהל קשר UDP (UDP connection) אחד על ידי ת'רד אחד המטפל בכל ה datagrams המגיעים לשרת, וכמו כן שולח datagrams.

ציין/ני, עבור כל אחת מהאפליקציות הבאות, את מודל השרת-לקוח המתאים לה ביותר ונמק'י זאת במשפט אחד. כמו כן ציין/ני עבור כל אפליקציה האם שימוש ב-pipelining מתבקש עבורה.

Time Server - הלקוח מתחבר ומקבל את השעה ב'מכונה' של השרת בפורמט של "HH:MM:SS".

Video Streaming Server - השרת משדר ב'זרימה' (streaming) סרטים לפי לוח זמנים (כמו בטלוויזיה). לקוח יכול להתחבר ולצפות בסרט המשודר כרגע על ידי השרת.

Database Server - הלקוח מתחבר, ומכאן ואילך שולח, להנאתו, מחרוזות המכילות שאילתות SQL, השרת מבצע את השאילתות על בסיס הנתונים ומחזיר את התוצאות ללקוח.

Web Server - לשם קבלת 'דף' (web page) הלקוח מתחבר לשרת שולח את כתובת ה URL של הדף, מקבל את תוכן הדף בפורמט HTML.

שאלה 7

(10 נקודות)

לקראת פתיחת הסינמטק בבאר שבע, החליטה ההנהלה לבנות מודל נתונים (data model) המתאר את מאגר הסרטים. בשלב ראשון הוחלט לאחסן את הנתונים הבאים:

- מספר קטלוגי לסרט
- שם הסרט
- שם הבמאי
- שנת הפקה
- אורך הסרט (בדקות)
- תאריך הקרנה (מחרוזת בפורמט dd/mm/yyyy)
- שעת הקרנה (מחרוזת בפורמט hh:mm:ss)

הערה: עקב משאבים מוגבלים, ניתן להקריין סרט אחד בלבד בכל נקודת זמן.

- א. תאר/י את מודל הנתונים המייצג את המידע הנ"ל באופן מנורמל (כלומר נתון יכול להופיע יותר מפעם אחת רק אם הוא מפתח). עליך להגדיר טבלאות ואילוצים. כל טבלה תוגדר בשרטוט, המכיל את שמה, המפתח הראשי והמפתח הזר (אם קיים כזה) שלה, ורשימת שדותיה.
- ב. קשר בין טבלאות יתואר על ידי מתיחת קו בין מפתח זר בטבלה אחת למפתח ראשי בטבלה שניה, תוך ציון מספר המופעים של המפתח בכל אחד מעברי הקשר. (5 נקודות)

שם טבלה:
מפתח ראשי:
מפתח זר:

ב. כתוב/כתבי שאילתת SQL המחזירה את נתוני הסרטים המוקרנים ביום פתיחת הסינמטק 1/2/2003. (5 נקודות)