class ClicksList
{
    // remove the first string in the list if equals to given parameter s
    // returns false if stop control ("S") was identified
    public synchronized boolean get(String s) throws InterruptedException
    {
        while (list_.size() == 0 ||
               ! (list_.getFirst().equals(s) ||
               ! list_.getFirst().equals("S"))
            wait();
        if (list_.getFirst().equals("S"))
            return false;
        list_.removeFirst(); // Remove first element of list
        notifyAll();
        return true;
    }
    ...
}

class UserInterface extends Thread
{
public void onStop() {
    try {
        clicksList_.add("S");
    } catch (InterruptedException e) {
    }
}

class Left extends Thread
{
    ...

    public void run() {
        while (true) {
            try {
                if (clicksList_.get("L"))
                    firemen_.moveLeft();
                else
                    return;
            } catch (InterruptedException e) {
            }
        }
    }
}

class Right extends Thread
{
    ...

    public void run() {
        while (true) {
            try {
                if (clicksList_.get("R"))
                    firemen_.moveLeft();
                else
                    return;
            } catch (InterruptedException e) {
            }
        }
    }
}
class Firemen
{
    ...

    public synchronized void moveLeft()
    {
        if (x_ > MIN_X) {
            synchronized (getThrower().getVictim()) {
                if (!(getThrower().getVictim().getX() = x_ &&
                    getThrower().getVictim().getY() = 1))
                    x_--;
            }
            repaint();
        }
    }

    public synchronized void moveRight()
    {
        if (x_ < MAX_X)
            synchronized (getThrower().getVictim()) {
                if (!(getThrower().getVictim().getX() = x_ &&
                    getThrower().getVictim().getY() = 1))
                    x_++;
                repaint();
            }
        }
    }

    ...
}

// Contents of x.h
#include <list>
#include <string>
#include "d.h" // class D
#include "c.h" // class C

class B;

class X {
```cpp
public:
    X( const C& );
    D Function1( int, char* );
    D Function1( int, C );
    B& Function2( B );

private:
    std::string  name_;  // Contents of x.h
    std::list<C> clist_;  // Contents of ximpl.h
    D d_;  // Contents of ximpl.h
};
```

```cpp
class XImpl {
public:
    std::string  name_;  // Contents of ximpl.h
    std::list<C> clist_;  // Contents of ximpl.h
    D d_;  // Contents of ximpl.h
};
```

**Question B (5 points)**

Class X contains includes XImpl, which is defined in another file and includes include files D.h and C.h. The class X is used to store data and include files. Includes are used to link the class with the other classes.

```cpp
// Contents of x.h
class XImpl;
class B;
class C;

class X {
public:
    X( const C& );
    D Function1( int, char* );
    D Function1( int, C );
    B& Function2( B );

private:
    XImpl* pXImpl_;  // Contents of ximpl.h
};

// Contents of ximpl.h
#include <list>
#include <string>
#include "d.h"  // class D
#include "c.h"  // class C

class XImpl {
public:
    std::string  name_;  // Contents of ximpl.h
    std::list<C> clist_;  // Contents of ximpl.h
    D d_;  // Contents of ximpl.h
};
```

**Question 3 (20 points)**

Class X contains includes XImpl, which is defined in another file and includes include files D.h and C.h. The class X is used to store data and include files. Includes are used to link the class with the other classes.
null destructor והDestructor מהפוך את הפעולה, לא וירטואלי, מה שוזמן🙌🏻 את המפעלה.

delete p;

הDestructor המחלקה Person:

virtual ~Person() {}
```cpp
const_cast<std::vector<int*>\&>(student.getCourses()).push_back
(new int(1200));
```

**She'ela 4**  
18 נקודות

<table>
<thead>
<tr>
<th>א</th>
<th>ב</th>
<th>ג</th>
<th>ד</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>V</td>
<td>X</td>
</tr>
<tr>
<td>X</td>
<td>V</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>X</td>
<td>X</td>
<td>V</td>
<td>X</td>
</tr>
<tr>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

**She'ela 5**  
12 נקודות

**ת療מו העיטוריה**

**ה Também**

1. מסר פיזיקל מובצעת על ידי מסר בר שלحلים מעל משאר מסגרת אוחד: thread pool של selector של מספר ב牺לשת המחשהالف.
2. התHdr של selector מובצע עזרה 999: ייחודי אוריינט על ה thread pool של selector.imir פיזיקל ב牺לשת המחשהالف.
3. קירית מבצעת על מסר בר של歷ם. שאר 999 התHdrים פיזיקליים למשרתם המחשה ב牺לשת המחשהالف. thread pool של selector.

**ת療מו تحتור**

thread pool

**ת療מו עיריה**

**ה También**
1. The task is to show the advantage of using thread pools over thread pools with additional limitations. In particular, the thread pools should be threadsafe, which eliminates thread contention.

2. A single thread can execute multiple tasks, unlike a single thread that executes only one task at a time.

The advantage of using thread pools is that they allow for efficient resource utilization, as threads are reused rather than created for each task.

Thus, the advantage of using thread pools is that they can handle multiple tasks concurrently, reducing the overhead of creating and destroying threads.

In contrast, using single-threaded reactors requires more context switching, which can be costly in terms of performance. Therefore, thread pools are preferred for applications with high concurrency demands.

For the third advantage:

1. Moving between threads involves context switching, which is expensive.
2. It is more complex than communicating between two reactors.

For the second disadvantage:

1. It is not possible to run the reactors immediately when the client receives a list of pairs: (host, port) under a list of ports.

3. The client should choose the best reactor according to the criteria of quantity and quality, considering the presence of more advantages than disadvantages.

4. The SQL query is as follows:

```sql
SELECT Analyses.POS, Analyses.Gender, Analyses.Number
FROM Analyses, Words, WordsAnalyses
WHERE Words.Str = WordsAnalyses.WordStr AND Analyses.ID = WordsAnalyses.AnalysisID AND Words.Str = 'מספרים'
ORDER BY WordsAnalyses.Prob Asc
```