Commonsense Computing: Concurrency and Concert Tickets

Gary Lewandowski
Xavier University

Dennis J. Bouvier
Southern Illinois University, Edwardsville

Robert McCartney
University of Connecticut

Kate Sanders
Rhode Island College

Beth Simon
University of California San Diego
Commonsense Computing: what students know before we teach

“You’re learning problem solving, not just programming.”

Students are not blank slates

Leverage the known

Exploit misconceptions as learning opportunities
Episode 3: Concurrency

Established literature – Ben-David Kolikant

Computer Science is more than programming

A chance to semi-replicate and compare
The task

Suppose we sell concert tickets over the telephone in the following way -- when a customer calls in and asks for a number (n) of seats, the seller
1) finds the n best seats that are available,
2) marks those n seats as unavailable, and
3) deals with payment options for the customer (e.g. getting credit or debit card number, or sending the tickets to the Will Call window for pickup).

Suppose we have more than one seller working at the same time. What problems might we see, and how might we avoid those problems?

66 students from five different U.S. institutions; type of institution varied
What we might hope students do

• Notice issue of multiple sellers and duplicate sales
• Notice issue of interleaving transactions
• Give reasonable (realistic) solutions
## Results by the numbers

<table>
<thead>
<tr>
<th>Description</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Identified selling seat more than once as problem</td>
<td>97%</td>
</tr>
<tr>
<td>Identified interleaving operations as a problem</td>
<td>9%</td>
</tr>
<tr>
<td>Provided a reasonable solution to the problem</td>
<td>71% students / 69% solutions</td>
</tr>
<tr>
<td>Brought up a different problem</td>
<td>41%</td>
</tr>
</tbody>
</table>
Describing the Synchronization Problem

“sellers could mark the seats unavailable at the same time” [ID412]
“there could be double booking” [ID106]

“one computer may be operating slower than another, causing the seats that one seller saw to be taken by another” [ID406]
“when and how long it takes to mark a seat unavailable” [ID410]

“The first, most obvious problem is that of overlap… the system might display to seller A that certain seats are open when … they have been reserved by seller B. …

Crossed signals. If seller A and B both book seats 145-160 …within milliseconds of one another, the instructions for reserving those seats on each of the other computers would cross mid-stream, introducing a problematic double booking or even worse, no booking at all.” [ID417]
What’s this “other problems” stuff?

“if the seats are marked unavailable before they are sold, the customer can change their minds before payment and ... hinder the sale ... to another customer” [ID417]

“...place their tickets at will call. It is possible for people to have the same name” [ID105]

“the seller not selling the seats efficiently to maximize the amount of people that can attend the event. ... Sometimes there will be a few seats not sold next to those groups of seats. [ID412]

“The computers may malfunction” [ID406]

Lots of ideas serve as distractors from what we see as the main problem.
Describing Solutions

“The program would have to temporarily mark seats that are being looked at … so that vendors couldn’t sell seats simultaneously” [ID313]

Centralized, implicit communication.

“set up the database so that only one person could access the database at a time.” [ID440]

“Instead of multiple people selling tickets and being involved in every step… the selling process [is] divided between two employees… While the second employee was taking care of the payment, the first could start to deal with the next sale” [ID120]

Centralized, explicit communication
“each vendor is responsible for a section of the concert hall” [ID303]

“sellers organize to sell specific seat sections. There can be an operator that finds out the general section that is desired, and forward the call to the seller of the section.” [ID 323]

Centralized, distributes resources to avoid needing synchronization.

“There should be some communication between the sellers. Ideally, the sellers would mark the seats as unavailable on the same documents, so that there would never be any doubling.” [ID101]

Decentralized: sellers communicate

“Use a computer program that networks each seller. This way, every seller has access to every seat available. As soon as a booking is made, it will automatically register on every seller’s screen and the chance of there being a double booking will be close to impossible.” [ID 323]
Errors and misconceptions

“… having a computer system that automatically (to the second) inputs the seat reservation” [ID438]

“we could have very high ‘refresh’ rates” [ID423]

“have the computer show the n seats as unavailable as soon as any seller has them pulled up on their screen.” [ID122]

“Each seller would have their own computer and all of them would be connected, so once a seat is claimed, all the other sellers will see it. If two sellers happen to click at the same time, a separate window will have to open and they both will have to try again.” [ID 402]
Compared to original study

• More solutions (55%) centralized in this study than Ben-David Kolikant’s (33%).

• Concentration on sellers rather than interleaving is consistent.

• Subjects in both studies simplified the problem in order to give solution.
What’s it all mean?

• Students can engage computer science questions beyond algorithms.

• Their concurrency analysis and solutions are in the ballpark of more experienced solutions.

• Students see many issues, need help discerning the key issues.

• Problems like this offer teachable points – real world situations that show the need for more CS knowledge.
Thank You 😊

• Data collectors Renee McCauley, Sara Miner More, Tammy VandeGrift, and Suzanne Westbrook.

• NSF: for inviting us to review grants during a really hot summer, and for paying for our work room at SIGCSE 06 and 07 via grant DUE 0243242.

• Umeå University: teleconferencing server!

• Josh, Sally, and Marian for helping us bond about CS Education Research in a meat-free way.
IT'S OVER!