



The value of developing soft skills by communicating with a general audience

Elliot Varoy, Mark Aziz, John Burrows, Oliver Sinnen and Nasser Giacaman

Parallel and Reconfigurable Computing Lab

The Traditional Programmer's Skillset

With the advancement of hardware and software technologies, technical knowledge remains an essential skill that employers require from graduates. However, as the software development industry moves towards large scale systems developed using agile methodologies interpersonal traits, or *soft skills*, become evermore important.

Students should be provided with the ability to develop communication skills wider than their immediate technical job description, a consistent concern identified by employers of graduates.

Educational Concepts

'Teaching is Learning'

It is well documented that students who participate in teaching others will develop better understanding of core concepts and organisation of their knowledge. Educational mobile applications can be built to take advantage of this, while maintaining a focus on improving communication skills. While students teach a non-technical audience, they will further improve their own understanding

'Sequential Learning'

The concepts involved in parallel programming involve many relationships and requirements of knowledge. This leads to the idea of progression of learning, or sequential learning, where specific topics must be taught before moving on to others. This concept should be utilised when developing educational applications to ensure students have the required prior knowledge before learning new concepts. A concept map for parallel programming concepts within the PDC is shown below.

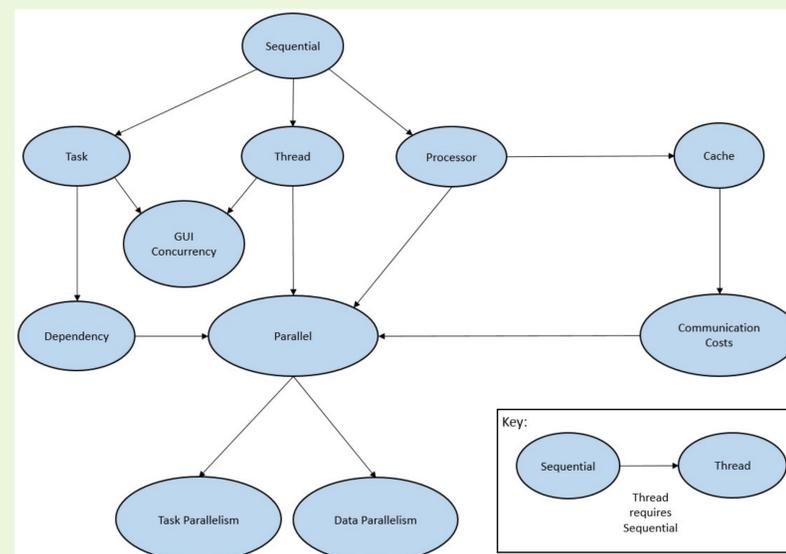
Parallel Patties – Educational Game

How does it work?

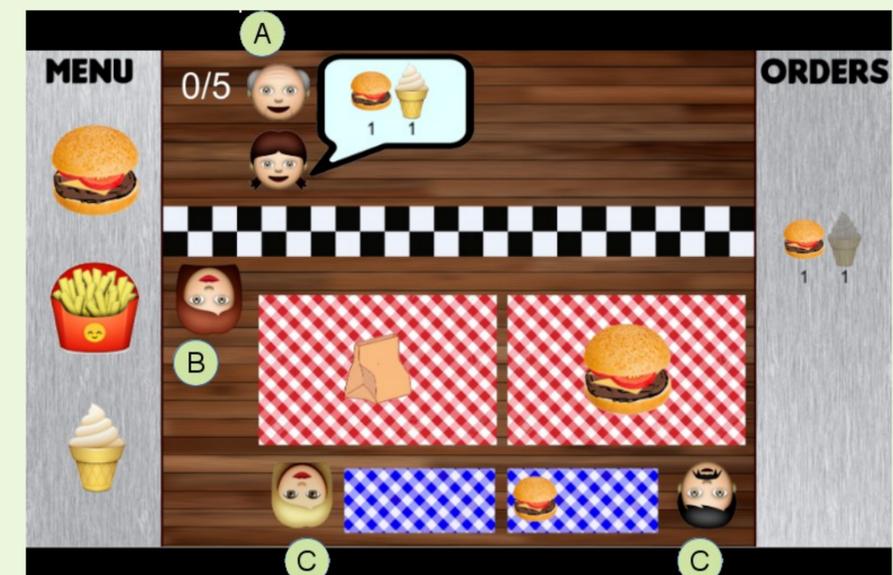
- Parallel Patties is a mobile application designed to utilise the benefits of analogies and gamification.
- Students are encouraged to use this application to communicate parallel concepts to a non-technical audience
- This application is an analogy for a burger shop, with many of its components representing specific components within a parallel system.
- The application consists of 6 levels, with each one focusing on a new parallel programming concept.
- Levels are preceded by a screen explaining the analogies within, to aid students in explaining the core concepts
- This will help introduce students to the skills required when explaining complex concepts to their non-technical peers, an important skill when entering large scale teams.



Sequential Programming with the GUI Concurrency Level. As level 2 in the application, it introduces the importance of delegating time-consuming tasks to a background thread (i.e. concurrency). The level before this does not include a background thread, causing a backlog of customer orders not being acknowledged (i.e. a non-responsive application).



Concept map outlining the relationships between a subset of core parallel programming topics. However, students are likely to “lose their audience” if they attempt to communicate these technical topics (in a technical manner) to a non-technical audience.



Screenshot of Parallel Patties, with the Task Parallelism Level. The red tables serve as processor cores, while the blue tables serve as caches. The small chef faces represent threads: “A” represents threads generating events (and therefore tasks), “B” represents the event-handling thread (vital for Conveying responsiveness in multi-threaded GUI applications), “C” represents the background worker threads.