

פרויקט בתכנות מקבילי ומבוזר 236371 - אביב תשע"ז

הרישום לקורס הינו רישום ידני בלבד ומוגבל מקום. כל הפרוייקטים דורשים קדם של מערכות הפעלה. רקע בתכנות מבוזר (דרך 236351 או 236370) הינו רצוי מאוד אך לא הכרחי.

על מנת להירשם לקורס, יש למלא טופס בקשת רישום הכוללת את שמות חברי הקבוצה, פקולטה, סימסטר, ורשימה מדורגת של 2-3 פרוייקטים מתוך הרשימה המצ"ב. יש להגיש את [הטופס מהלינק](#) [הבא](#) עד יום שלישי 28/2/2017 בצהריים (12:00). תשובות ינתנו עד יום רביעי ה-15/3/2017.

רשימת פרוייקטים בקורס בתכנות מקבילי ומבוזר:

RUST projects

Rust is a new programming language that offers novel programming concepts with low memory footprint and is well suited for developing distributed systems.

Memory management is manual, but is much simpler and safer than with C/C++.

These benefits has lead Dropbox, for example, to base their novel highly scalable storage system on Rust!

1. **Multi-Paxos in Rust.** Implement the multi-Paxos protocol for replicated state machine in Rust. The implementation should be generic enough to suite as a general purpose library to be run on multiple processes, whether they are in the same machine or distributed in the same LAN or even across the Internet. The end goal is an open source distribution in github or an equivalent. Suitable for 1-3 students - the scope will be adjusted to the number of students.
2. **PBFT in Rust.** Implement a PBFT library for Byzantine tolerant block chain replication in Rust. The implementation should be generic enough to suite as a general purpose library to be run on multiple processes, whether they are in the same machine or distributed in the same LAN or even across the Internet. The end goal is an open source distribution in github or an equivalent. Suitable for 2-3.

Other projects

3. **Heavy hitters identification using Open vSwitch (OVS) and Storm.** Open vSwitch is the leading open source implementation of the Open Flow standard for Software Defined Networks (SDN). Heavy hitters identification is an important capability for various networking functionalities such as load balancing, quality of service enforcement, and distributed denial of service (DDoS) attacks identification. In this project you will implement a hierarchical heavy hitters (HHH) identification algorithm recently developed in our department. The goal is to export data routed by OVS to a storm cluster whose bolts implement an HHH algorithm and try to optimize this configuration for best performance. Suitable for 2-3 students.
4. **TinyTable in C/C++.** TinyTable is a compact hash table that can be used to efficiently maintain networking flow statistics. It is documented in the following report: <http://www.cs.technion.ac.il/users/wwwb/cgi-bin/tr-info.cgi/2015/CS/CS-2015-04>. Its functionality is very important for networking devices, but can also be used as a

general purpose highly space efficient hash table. We currently have a Java implementation. In this project, you are to implement TinyTable in C/C++. The implementation should be as efficient as possible. Suitable for 1-2 students.

5. **TinySet in C/C++.** TinySet is an access efficient Bloom filter construction. It offers better performance and memory locality than standard Bloom filters, which are used in networking devices, distributed storage systems, pub/sub, Internet browsers, and many other applications. TinySet is documented in the following: <http://www.cs.technion.ac.il/users/wwwb/cgi-bin/tr-info.cgi/2015/CS/CS-2015-03>. We currently have a Java implementation. In this project, you are to implement TinyTable in C/C++. The implementation should be as efficient as possible. Suitable for 1-2 students.
6. **Food Helper.** You are in the supermarket and suddenly see that zucchinis are on sale. But, you don't know what to do with them. Or, you open your fridge and all that is left is cabbage - can you turn this into a nice dinner? Simply snap it with your phone and wait for your friends to send you potential recipes. An important aspect of this project is on having an intuitive and nice UI. Ideally, it should integrate well with an existing social network platform. Additions can be, for example, ranking your friends suggestions and seeing how their friends have ranked them in the past to help you choose the most promising recipe. Also, think of how the list of ingredients should be best handled. Suitable for 2 students.
7. **Augmented Reality Mobile Game.** The goal here is to develop an augmented reality game played with a mobile phone. Ideally, it should involve the phone's camera, GPS, and Internet connectivity. Social interaction is a major plus. Be creative! Suitable for 2-3 students.
8. **Networking stream processing.** The goal here is to implement a recently developed network stream processing algorithm in C/C++ and perform timing measurements. The implementation should be as efficient as possible. Suitable for 1 student.

הנחיות כלליות

בכל הפרוייקטים, בציון הסופי יילקחו בחשבון ההיבטים הבאים:

- מסמכי ביניים: functional specification, high-level design (10%)
- מסמכי הגשה סופית: user manual, programmer's guide, documentation (15%)
- מצגת סופית (10%)
- איכות ה"מוצר" של הפרוייקט עצמו (55%)
- מקוריות וחדשנות מעבר לדרישות הבסיסיות של הפרוייקט (10%)

במהלך הסמסטר יתקיימו 2 פגישות מעקב (שבוע 2, שבוע 5), גירסת אלפא (שבוע 12) והגשה סופית. יש להגיש את מסמך ה-functional specification עד למפגש המעקב הראשון, ואת מסמך ה-high-level design עד למפגש המעקב השני. שאר המסמכים יוגשו ביחד עם ההגשה הסופית.