Dear I-SURF Fellows, Faculty Mentors, and Guests:

The pursuit of knowledge is a universal goal, but it can take on different forms in different places. With that in mind, students often look for opportunities to enhance their education by engaging in different approaches to learning, leadership, entrepreneurship and studying within other languages and cultures.

This summer, the International Undergraduate Research Fellowship (I-SURF) brought 15 undergraduate researchers to UC Irvine from Kookmin University in Seoul, South Korea. They were selected to conduct software development research under the guidance of five nationally-distinguished faculty mentors from the Donald Bren School of Information and Computer Sciences. Students chose from a variety of challenging and original research projects, each exploring a diverse and exciting range of topics within the field of computer science. Faculty mentors and their teams of graduate students and post-doctoral fellows provided personalized mentoring and training to the I-SURF Fellows, giving them the unique opportunity to explore the future of computer science research, learn about how research is conducted, and to become immersed in UC Irvine’s collaborative research culture.

The I-SURF Fellows dedicated themselves to full-time work on their research projects throughout the program. In addition, the students explored their own futures as well, looking into the vast array of possibilities that lie before them. They attended seminars on a wide variety of topics relevant to their research and took an intensive course on technical communication. They toured state-of-the-art labs, local industries and successful startup companies in Silicon Valley.

The 2015 I-SURF Program highlighted the successful collaboration between Kookmin University and the Undergraduate Research Opportunities Program (UROP), at UC Irvine. UROP is committed to supporting faculty-mentored undergraduate research and creative activities in all disciplines. In addition to sponsoring the I-SURF program, UROP also advises undergraduate students about on- and off-campus research opportunities, and publishes The UCI Undergraduate Research Journal, an annual multidisciplinary publication. In addition, the Summer Undergraduate Research Program (SURP) provides students with the opportunity to immerse themselves into a research project or creative activity under the guidance of UC Irvine faculty members. UROP has also collaborated with other units on campus to sponsor undergraduate research programs emphasizing multidisciplinary design, as well as research in biophotonics, health promotion and disease prevention, information technology, cardiovascular research, chemistry, and micro/nano technologies.

Thank you for participating and for showing your support for the I-SURF Fellows presenting here today. I also would like to offer our thanks to Professor Sung-Soo Lim of Kookmin University, who has been a vital part of the students’ UC Irvine experience. We look forward to many more years of collaborating with him and his colleagues. Finally, a special note of appreciation goes out to the faculty mentors who have devoted much time and effort mentoring these students. We look forward to following up with the continued achievement of these outstanding individuals, and hope that you leave today’s program inspired by their efforts and enthusiasm.

Sincerely,

Said M. Shokair
I-SURF Director
Director, UROP
## Schedule of Presentations

Group presentations are allotted 20 minutes followed by a 10-minute question and answer period. Individual presentations are allotted 15 minutes, with 5 minutes afterwards for questions and answers.

Electronic copies of Students’ PowerPoint presentations and abstracts are available on the I-SURF Web site, [http://www.u rop.uci.edu/i-surf.html](http://www.u rop.uci.edu/i-surf.html). Click “Participants,” then the name of an individual student.

### Wednesday, August 26, 2015

**Calit2 Auditorium**

<table>
<thead>
<tr>
<th>Time</th>
<th>I-SURF Fellows</th>
<th>Project Title</th>
<th>Faculty Mentor(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>9:00</td>
<td>Welcome</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9:15</td>
<td>Hyeongjun Lee</td>
<td>UCI Smart Campus</td>
<td>Elaheh (Eli) Bozorgzadeh Computer Science</td>
</tr>
<tr>
<td></td>
<td>Aram Oh</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Jiung Shin</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9:45</td>
<td>Kisang Cho</td>
<td>Power Consumption Minimization for Android-Based Smartphones</td>
<td>Nikil D. Dutt Computer Science</td>
</tr>
<tr>
<td></td>
<td>Hoyeonjiki Kim</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Jihyun Park</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10:15</td>
<td>Wonbeom Choi</td>
<td>Executable Trace Tool</td>
<td>Ian G. Harris Computer Science</td>
</tr>
<tr>
<td></td>
<td>Sunghyun Hong</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Yunho Kim</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10:45</td>
<td>15-Minute Break</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11:00</td>
<td>SeulGi Choi</td>
<td>WebRTC Open Source Project Enhancement and Related Integrated Online Service and Product Development</td>
<td>Alexandru (Alex) Nicolau Alexander Veidenbaum Computer Science</td>
</tr>
<tr>
<td></td>
<td>SeongJung Kim</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>DongUk Lee</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11:30</td>
<td>Yoonseung Choi</td>
<td>How to Survive Out-of-Memory Errors</td>
<td>Guoqing (Harry) Xu Computer Science</td>
</tr>
<tr>
<td></td>
<td>Soyeong Park</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12:00</td>
<td>InCheol Lee</td>
<td>How to Reduce the GraphQ’s Pre-Processing</td>
<td>Guoqing (Harry) Xu Computer Science</td>
</tr>
</tbody>
</table>
Project Title: UCI Smart Campus

Faculty Mentor: Elaheh (Eli) Bozorgzadeh, Computer Science

Abstract:
Outdoor smart spaces are equipped with various electronic devices—smart sensors, cameras, charging stations, IoT devices, and interactive displays—that are distributed widely throughout the area. Using the power grid or batteries to power all these devices is not a scalable option. One promising solution is to use renewable energy such as solar energy harvesting to power such devices. In order to achieve this, we propose to generate a harvesting map of a given space to evaluate the harvesting potential of the space and identify locations for solar panel installations, charging stations, etc. We have analyzed shadow effects from buildings, clouds and the path of the sun. This project provides information on potential solar energy outdoor points of interest, such as outdoor café sitting areas and outdoor social settings on the UC Irvine campus map. As a result of this process, students not only can see an estimate of the current day’s solar energy but also get sun and shadow data for locations throughout the campus.

We have developed a software framework at both server and mobile platform levels to demonstrate the solar energy map while considering shadow effects from buildings and clouds.
**Project Title:** Power Consumption Minimization for Android-Based Smartphones

**Faculty Mentor:** Nikil D. Dutt, Computer Science

**Abstract:**
Games have become one of the most popular applications for the mobile platforms. Most games, however, consume more power and drain the platform batteries faster than other applications. Recent platforms are equipped with Heterogeneous Multiprocessor Systems-on-Chip (HMPSoC), which integrate central processing units (CPUs) and graphics processing units (GPUs) on the same chip. This configuration enables high-end gaming on the platform but with the cost of high power consumption. To solve this problem, this research aims to minimize energy consumption with minimum compromise on frames per second (fps) by using two approaches. Default run-time power management for these platforms is performed somewhat independently for the CPU and GPU. CPU power management is performed by a standard Linux power manager and GPU power management is handled by closed source firmware. These power managers operate outside an application’s quality of service (QoS). Thus, they lack synergy and may waste power by overclocking the CPU and/or the GPU. The first approach solves this problem by creating an integrated CPU-GPU governor. The second approach is focusing on the correlation between platform hardware characteristics and power/performance metrics. Each characteristic has different effects on power consumption and performance. So by determining the correlation, this approach can create several opportunities for a new policy design which will include a new governor algorithm.
**Project Title:** Executable Trace Tool

**Faculty Mentor:** Ian G. Harris, Computer Science

**Abstract:**
Malware is common throughout the Internet and it is important to detect these abnormal programs easily. Also, the process of debugging software to look for the effects of malware can be extremely tedious. To solve these problems, we have created a tool to gather debugging traces and learn how input value impacts control flow. Our research aims to implement an executable trace tool by using the “IDA PRO” and “Immunity Debugger” disassembly programs. These programs create x86 assembly code and the tool can run PE files compiled by the GCC compiler only. When the program makes assembly code, it automatically traces the code line by line by using the python script - pycommand. When the executable trace tool searches branch instructions, it compares the current address with the next address. From this result, the tool determines whether a branch value is true or false and creates a raw branch file. The tool also counts how many times each local function is called and creates a raw function call file. Based on these raw files, the executable trace tool creates statistics files of branch instructions and function calls. If a user has statistics files from normal execution file, they can be compared against a malware-infected program. Also the results can be displayed graphically to help developers can find bugs more easily.
Project Title: WebRTC Open Source Project Enhancement and Related Integrated Online Service and Product Development

Faculty Mentors: Alexandru (Alex) Nicolau & Alexander Veidenbaum, Computer Science

Abstract:
Common video chatting apps require users to install additional programs to use their service. Also, service providers need a signal server to allow users to share basic information with one another, and a storage server that stores users’ communication history. This project helps users do video chatting without any installation and it helps providers to supply video chatting service without servers by using WebRTC (Web Real-Time Communication) and the Omlet messaging platform. WebRTC is a web standard that was drafted by W3C, so it works on any web browser and allows them to do peer-to-peer communication. Omlet provides an environment for developers as an API, which allows developers and customers to use it for free. We used these two tools to create the OmletRTC API, which helps develop a WebRTC app on the Omlet platform. OmletRTC is superior to common video chatting app in accessibility, cost and security because customers can use it simply and providers can support it without the cost of setting extra servers. Moreover, because it communicates directly between clients, it ensures users’ privacy.
Project Title: How to Survive Out-of-Memory Errors

Faculty Mentor: Guoqing (Harry) Xu, Computer Science

Abstract:
As modern computing enters the era of Big Data, Hadoop MapReduce has become a widely-adopted programming model for large scale data processing due to its simple interface. However, data-intensive applications often fail to achieve satisfactory performance and/or suffer program crashes due to out-of-memory errors. We have conducted an extensive study by gathering and analyzing real-world performance cases submitted by StackOverflow users. The study shows that performance problems are common in Big Data systems. We also find that while there are many reasons for this poor performance, practical solutions always center on making “best practice” recommendations for tuning framework parameters. Unfortunately, a framework such as Hadoop has numerous parameters that interact in complicated ways. Our research aims to identify the parameters that cause the greatest impact and understand their contribution to the applications’ performance. We developed a set of in-house data-intensive programs and several real-world business applications. These applications were then heavily simulated under various configurations in a cluster. We have found two important parameters and performed rigorous stress tests on them. Our experimental results provide strong empirical evidence for a near-optimal configuration under which many Big Data applications may have better performance and/or survive crashes. We also have distilled valuable insights that can aid non-expert developers in their performance tuning process.
Project Title: How to Reduce the GraphQ’s Pre-Processing

Faculty Mentor: Guoqing (Harry) Xu, Computer Science

Abstract:
Big Data Processing is among today’s hottest computer technologies, and has spread rapidly through the Internet technology industry since the beginning of 2012. The goal of this project is to conduct graph query processing of big data using the GraphQ framework, a scalable querying framework for very large graphs. GraphQ is built on a key insight that many graph properties can be computed effectively by exploring only a small fraction of the graph and that traversing the complete graph is not necessary. The centerpiece of our framework is the novel idea of abstraction refinement, where a very large graph is represented at multiple levels of abstractions, and a query is processed through iterative refinement across graph abstraction levels. As a result, the GraphQ framework has several distinctive traits unseen in existing graph processing systems. With GraphQ, a wide range of complex analytical queries can be answered over very large graphs with the resources available on a single PC. Recently, GraphQ’s pre-processing routine has been determined to be inefficient. Therefore, I have remodeled GraphQ’s pre-processing sections by optimizing code to increase efficiency. My role has been to reduce pre-processing time by optimizing code. Our research suggests three potential methods to do this: reducing the amount of loading, considering the graph size, and using a bit array. Applying these methods will help in improving the efficiency of the graph algorithm.
Thanks & Acknowledgements

Many thanks to the following individuals, institutions and groups for their support in making the 2015 I-SURF Program an outstanding success.

Chancellor Howard Gillman

Provost and Executive Vice Chancellor Enrique Jose Lavernia

Vice Provost and Dean Michael B. Dennin, Division of Undergraduate Education

Dean Hal S. Stern, Donald Bren School of Information and Computer Sciences

South Korea
Ji Soo Yu, President of Kookmin University
Sang-Hong Lee, President of IITP
Kwangsoo Hahn, Dean of College of EECS, Kookmin University
Dohyeon Kim, VP of Office of Entrepreneurship, Kookmin University
Eun-Jin Im, Chair of School of Computer Science, Kookmin University

Faculty Mentors
Elaheh (Eli) Bozorgzadeh, Computer Science
Nikil D. Dutt, Computer Science
Ian G. Harris, Computer Science
Alexandru (Alex) Nicolau, Computer Science
Alexander Veidenbaum, Computer Science
Guoqing (Harry) Xu, Computer Science

Additional Thanks to:
Anteater Recreation Center (ARC)
Arroyo Vista Housing
Associated Students of UCI (ASUCI)
Budget Office
Calit2 Team
Division of Undergraduate Education
Donald Bren School of Information and Computer Sciences
Graduate Students and Post-Doctoral Fellows
Hospitality & Dining Services
International Center
Office of Global Engagement
Summer Sessions
UROP Team

If you would like further information on the I-SURF Program, please contact the Program Director:

Said M. Shokair
I-SURF Director
Director, Undergraduate Research Opportunities Program (UROP)
Division of Undergraduate Education
Student Services II, Suite 2300
University of California
Irvine, CA 92697-5685
Phone: (949) 824-4189
FAX: (949) 824-1607
E-mail: shokair@uci.edu