Hello and welcome to our 2012 edition of the CS program newsletter! In the past year, we have seen quite a few new and exciting developments. Most importantly, our enrolment is up – again! As if this would be even possible, our enrolment is up again by 16% compared to last year and up by 34% compared to Spring 2010.

Along with these increases we have seen more and more students choosing our Applied Computer Science program, please see my report on this program in last years newsletter. Both of our programs are more popular than ever!

Eventually, we now start experiencing our steady enrolment growth in our upper division classes and the senior capstone experience class. In this year’s capstone class, we have six teams with four to five students each! That said, we are in constant demand for interesting and challenging capstone projects. To become a capstone sponsor now is also going to be even more rewarding. We are in the process of switching over from a one-semester capstone class to a two-semester capstone sequence, starting in the fall. This means that students will have more time to work on a project, ultimately resulting in higher-quality project outcomes. Don’t hesitate to become a capstone sponsor and get that pet project done that you have been thinking about for a while. It’s easy, just drop me an email!

Last year we have also seen quite some turnover in our team. Last summer two of our tenure-track faculty (Dr. Li and Dr. Wang) left us to pursue career opportunities elsewhere that better fit their skill sets and interests.

Abe Pralle, our most popular lecturer, who has been on our team for many years, has also left us to dedicate more time to his gaming company Plasmaworks (www.plasmaworks.com). Abe was very active in gaming-related research, resulting in a production-quality gaming environment that he also used in his classes. He leaves a big gap, which will be difficult to fill. We wish him all the best for his future endeavors!

Please visit our new web site:
http://nau.edu/cefns/engineering/computer-science/
On the other hand we have a new faculty member on board of our team. Dr. Gregory Caporaso was hired to establish a bioinformatics track here at NAU. We are really excited about this incredible opportunity! Our first pilot semester went very well; our students are very receptive to this new program track. You will find an article by Dr. Caporaso below that reflects on his experiences during his first year.

Finally, Dr. Eck Doerry has extended his commitment to international education. He is leading the GSEP initiative that he brought to life from the ground up (please see his interesting article below). As an effect, his teaching capacity is cut into half.

As an effect of all the changes in personnel we are running two searches right now, one for a tenure-track position and another one for possibly two lecturer positions. We are very optimistic to find exactly the people that fit our department best!

We have also been working hard to provide our students with the best learning environment possible. We put our class fee monies to work and equipped our senior projects room with top-of-the-line-Macs. We also considerably updated our thin client-based computer lab to significantly decrease response times and improve the overall experience. Finally, we remodeled our ACM clubroom from scratch. It now has a fridge and nice wood-finish cabinetry. Some students are known to stay there overnight …

I hope you enjoy our newsletter! If you are ever in the area or passing through Flagstaff, please drop by. We are always happy to reconnect with our friends and alums and our ACM student club is always looking for speakers to share insights.

All the Best,
Wolf-Dieter Otte, Associate Chair and CS programs coordinator

---

**STUDENT CHAPTER OF ACM**

NAU ACM is the NAU chapter of the Association for Computing Machinery. Computer professionals and enthusiasts are welcome to join, where they will learn about fascinating new technologies and gain a deeper understanding of existing technologies. Members of ACM have easy access to tutoring and study materials, and have a quiet area conducive to studying and completing assignments. ACM also provides unique opportunities for members, such as building and programming a robot, creating video games for the ACM game cabinet, and competing in national programming competitions.

Weekly ACM meetings are open to everyone, and are held in room 105 of the engineering building at 6:00 every Thursday. See [http://www.nau.edu/acm](http://www.nau.edu/acm) for more details!
The LEGO Mindstorms NXT combines Technic pieces that are used to construct the physical structure of robots with a microprocessor, sensors, and servo motors that can be mounted on this structure in order to create robots for a wide variety of applications. Data can be collected using a variety of sensor types: For this project, students may use sensors for gathering information about light, sound, touch, a number of ultrasonic readings, color, as well as a compass sensor. Programming the NXT Intelligent Brick, a 32-bit ARM microprocessor with 64KB of RAM, takes place using the Java programming language. Compared to other commercially available robot-building kits, Mindstorms has the advantage of being low-cost and easily reconfigurable to support the creation of new kinds of robots.

Each student team is composed of three to four software engineering students that collaborate to design a robot that can successfully complete a challenge course. The goal of the robot is to successfully navigate a playing field while staying within that field’s boundaries, scan this playing field for objects of a specific color, retrieve red objects, and return these objects to the starting location without disturbing the location of any other objects. The problem is deceptively simple, as there are significant challenges that must be solved while designing and implementing the control systems of these robots: sensing objects and identifying colors, obstacle-avoidance, and mapping of the playing field so that accurate navigation can take place. While projects are still under way as of the time of this writing, the figure below shows the physical structure of one team’s robot.

Student teams are responsible for completing a collaborative design effort that includes elaborating on the software requirements of the control system, creating an appropriate software architecture, implementing the control system in Java, and creating a suitable test plan to support verification. Student teams – despite documenting their requirements and design – use an agile development process and tools supporting collaboration and source code version control, like ScrumDo and Github. Furthermore, students also give presentations on their designs and progress, which supports the development of important communication skills and is intended to better prepare them for their capstone experience.

**Liang Chen Joins NAU CS for Spring 2012**

Meet Liang Chen. She is a visiting scholar from China working with NAU computer science faculty. Ms. Chen is a lecturer in the School of Computer Science, Xi’an University of Posts & Telecommunications, Xi’an, Shaanxi, China. Take a look at her beautiful campus at this web site: [http://www.xiyou.edu.cn/english/](http://www.xiyou.edu.cn/english/)

Ms Chen’s research interests include operating systems, the C programming language, and embedded systems. She also enjoys Linux. She teaches classes in C, operating systems, and Linux apps.

Ms. Chen is presently auditing CS110 (Introduction to Computer Science), CS126 (CS1), and CS460 (Computer Networking). She regularly participates in CS program area faculty meetings, and also meets weekly with NAU’s Center for International Education.

She has made some significant observations of CS education here vs. at her university in China. For example, she has noticed we give the students more time to explore concepts with self-directed homework and projects. She says her courses at Xi’an University are more highly structured, with very specific lab sessions and classroom content. She says her classes have a heavier weighting of the final exam, while ours often count higher both class projects and homework. She also said Xi-an has bigger class sizes, and she can sense how the NAU students appreciate getting to know their professors in smaller-sized classes.

Ms. Chen appreciates the opportunity to continue to increase her English language skills, while observing NAU teaching styles in the classes she is auditing. When she returns, she will be expected to take on more bilingual students. She will be better equipped to aid in their success.

Ms. Chen applied for this opportunity over a year ago. She is partially supported by NAU’s 1-2-1 educational program with China, her employer in China (Xi’an University), with the rest at her own expense.

Ms. Chen is only here for spring semester 2012, so, please make an effort to contact her and introduce yourself. She can be reached at chen.liang@nau.edu
Bioinformatics is an emerging cross-disciplinary field that focuses on the application of the tools of Computer Science to address questions in Biology and Medicine. Bioinformatics is emerging as a field of study largely as a result of the large influx of data in nearly all areas of biology. This is exemplified by the decreasing cost of DNA sequencing (i.e., determining the linear order of nucleotides in a strand of DNA), which exceeds the rate of the decreasing cost of computing (Moore’s Law). For instance, a single 10-day run of the Illumina HiSeq2000 DNA sequencing instrument generates over 100GB of data. Traditionally trained biologists generally don’t have the skills to effectively work with this quantity of data, and as a result there is an increasing need for students trained both in biology (to understand the systems under investigation) and computer science (to understand the computational tools that are necessary to facilitate the next round of advances in Biology).

I joined the Computer Science faculty at NAU in Fall of 2012 to initiate a training program in bioinformatics that crosses the traditional disciplinary boundaries. This program is off to an exciting start: this year I designed and taught two classes, Bioinformatics I and II, which are cross-listed in Computer Science and Biology and have succeeded in attracting undergraduate and graduate students from these departments.

The structure of this Bioinformatics training program is still in development, but it will likely become a minor or a certificate-granting program. After initial prerequisites (Computer Science I and General Biology I) students from different majors will begin together on a three-course progression: Bioinformatics I – III. The educational goals include a working knowledge of the python programming language (which is widely used in bioinformatics and scientific computing in general), molecular biology, and interacting with remote Unix/Linux systems. The latter is extremely important as many bioinformatics analyses are run on remote compute clusters. Additionally there is a lot of interest in cloud computing in bioinformatics to support large-scale data analysis by groups without in-house compute clusters (e.g., researchers in small biology labs or developing nations, or students in high schools or community colleges). To help students obtain experience in cloud computing, I received a grant from Amazon Web Services (AWS) for the two Bioinformatics courses taught this year, and students are learning to interact with remote Linux systems in this context. Finally, a major goal of these courses is to introduce students to working together in interdisciplinary groups. To achieve this goal, the courses become increasing project-oriented throughout the progression, and groups are always designed to consist of students from both majors. The projects focus on diverse and timely areas of bioinformatics: this year the projects have focused on techniques for analyzing the human microbiome, the (mostly beneficial) microbes that live in and on our bodies and play an integral role in our health. The human microbiome is currently a very active area of research, and is regularly the subject of articles in top journals such as Science and Nature.

My teaching website provides additional information on the nascent Bioinformatics training program at NAU: http://www.cefns.nau.edu/~jgc53/teaching This was an exciting year for Bioinformatics at NAU, and I suspect that next year will be even more so as this program continues to take off!

http://www.caporaso.us

The GSEP initiative: Computer Science leads NAU towards global competency

by Dr. Eck Doerry

The Rapidly increasing globalization has been the dominant force shaping the development of the world economy in the past decade. Modern practice in engineering and science has been at the forefront of this trend, with nearly all mid- and large-sized companies in these sectors acquiring international facets in the areas of design, product development, manufacturing, and marketing. As a discipline, Computer Science has been particularly active in exploring global options, simply because our product — software codebases — can so easily be accessed and developed by a widely-distributed team. No impellers, circuit boards, or other hardware to ship around the world here; software collaboration is a simple matter of connecting to the version control server. As a result, it is now very common for software development teams to be composed of a global mix of talent, and new graduates must be prepared to work effectively across national, linguistic, and cultural boundaries in order to compete for the best jobs.

This year, I’ve taken on a special project to make sure our science and engineering graduates are ready for the global marketplace: the new Global Science and Engineering Program (GSEP) provides motivated science and engineering majors an “upgrade option” to get more out of their education dollar and prepare them to compete for the top jobs in tomorrow’s global economy. After five years of study in this dual-degree curricular track, GSEP graduates will emerge with their regular B.S. degree in their chosen engineering or science major, plus a second B.A. degree in a modern language and an International Engineering and Science Certificate. More important than any diploma is the unique internationalized professional preparation the program provides: GSEP scholars spend their fourth year entirely abroad. In the first semester abroad, students complete key coursework for their NAU degree at a partner institution abroad; during the second semester, GSEP scholars are placed in a professional internship at a company or laboratory, where they will have a chance to
apply both language and discipline-specific skills in a real-world context.

As a specific initiative targeted at producing more globally competitive science and technology specialists, GSEP is accessible only to majors in our college (Engineering, Forestry and Natural Sciences; CEFNS). I’m happy to say that Computer Science is leading this exciting initiative, not just because I’m developing it, but also because of the unusually strong response from our computer science majors: 8 of the 43 current participants are CS majors, by far the highest participation of any CEFNS program, as a percentage of total majors. The elite, challenging nature of the GSEP curriculum also tends to attract particularly gifted, motivated students; the average GPA of GSEP participants is 3.62 with SAT scores averaging above 1760. Wow!

We expect GSEP to continue its steep growth going forward, drawing exceptionally gifted students from across the nation, and becoming another of the unique signature undergraduate programs that NAU is known for. To find out more about the GSEP, visit the GSEP website at: http://international.nau.edu/GSEP/

I want to close with a call for assistance to all of our alumni out there: clearly the international internship is the most valuable training experience in GSEP, the apex of the whole program. At the same time, finding these internship opportunities for GSEP scholars represents the greatest challenge I face. **Maybe you can help:** if your professional life includes contacts (corporate divisions, partners, affiliate companies, etc.) in international locales speaking German, Spanish, French, Chinese, or Japanese that you think might be able to provide some internship opportunities, please let me know. Personal connections to potential internship providers are so much more effective than cold calls…and the high quality of GSEP participants makes it a win-win proposition for both sides!

**Thanks!**

---

**Introducing Computer Science with Python**

**by Dr. James Palmer**

At NAU we pride ourselves in cultivating exceptional computer scientists and software engineers. This begins with the introductory sequence of classes that our students take: Computer Science I (CS1), Computer Science II (CS2), and Data Structures (DS). Many students unfamiliar with the discipline are surprised that we spend just as much time in CS1 and CS2 talking about problem solving, design, architecture, documentation, and testing as we do about programming language syntax and semantics. That’s because our intro sequence isn’t designed to merely teach students a programming language so much as it’s designed to teach students to think like a computer scientists and solve problems with computation.

Currently we use Java in CS1, CS2, and Data Structures but in the fall of 2012 we will start with Python in CS1 and end with Java in Data Structures. The reason NAU and many other universities have historically started with Java is that it’s not a bad introductory language. Java has a relatively simple syntax, supports object-oriented programming, and generally promotes very high-level design. It has been a good language that has provided a foundation for learning other languages.

But Java isn’t without its flaws – compared to more recent languages like Python, JavaScript, and Ruby, Java is not nearly as dynamic and concise. Such languages often have higher level programming interfaces for graphics, sound, user interfaces, and web-based applications. These higher level APIs let us approach more complicated problems earlier in the curriculum. And getting to the problem solving quickly is important in CS because it both motivates students and reveals what our discipline is really about.

NAU faculty are not alone in this assessment. A host of papers in pedagogical conferences such as CCSC, SIGCSE, ASEE, and ITICSE have examined using Python in the classroom emphasizing the language’s approachability and high level graphics, audio, web, and gaming APIs. Book sales by Pearson have also shown year-to-year double-digit sales growth of Python textbooks. And outside academia, the growth of dynamic languages such as Python, PHP, JavaScript, and Ruby is changing the way we do application development.

This is exemplified in bioinformatics applications that use this linguistic dynamism and power to connect scientists with high-level APIs as a mechanism to explore problems. Dr. Greg Caporaso, a new assistant professor in our program, is the lead developer of the QIIME project, which provides a python based bioinformatics package for analyzing microbial communities. Dr. Caporaso has already developed two bioinformatics classes that utilize Python. These classes serve both Computer Science students and Biology students at NAU.

Thus, transitioning CS1 and CS2 to Python not only serves our pedagogical goals but will also support a developing bioinformatics track in our program where bioinformatics students take both CS1 and BIO1 as prerequisites for Dr. Caporaso’s bioinformatics classes. Java will continue to serve a role in our Data Structure class where it’s a much better choice with its static typing and explicit structural formalisms. We will also continue to offer broad exposure to other major languages and programming paradigms throughout our curriculum because when it comes down to it we don’t teach Java or Python, we teach Computer Science.
CAREERS

BY STEVE JACOBS, ADJUNCT FACULTY

Computer Science majors are again heading into an excellent career. The “Best Careers in 2012” lists from U.S. News and World Report, CNN, and others point to computer science, software engineering, and related careers (such as IT) as still being in demand. Often these careers appear on lists of well-paid entry-level positions.

I encourage students to create and maintain a current resume. Make sure you take advantage of NAU-provided resources to aid you in your job search, such as seminars (resume writing, interview techniques, etc.) and campus recruiting events. Get a hold of the “Job Search Guide” from the NAU Gateway Student Success Center.

Attend NAU on-campus Career Fairs, even if you are not graduating until much later. Often, employer representatives can give insights as to courses they need you to complete, while they share any internship opportunities.

And, be flexible. Keep in mind many of the jobs may not be precisely what you hoped for – for example, you may be asked to develop software in C++ when you really wanted to do Python, or you may be asked to do some web programming when you really wanted to build database applications. You may be asked to maintain (i.e. enhance or improve) existing, operational code. The key to success in the marketplace is flexibility both in assignments and work locations.

Lastly, keep in mind your learning does not stop at NAU. You will be learning new skills and development processes on the job.

NEW INSTRUCTOR PERSPECTIVE

BY RYAN MIDDLETON, INSTRUCTOR

As a new instructor for the department I look forward to meeting and teaching new students as they enter into the program and progress through the classes that I teach. Having completed the Computer Science program here at NAU, followed by several years of industry experience, I understand the learning environment provided by NAU and what makes it “the difference that matters.”

I finished the B.S. Computer Science program here at NAU in 2005 and spent the next several years in industry working for Southwest Windpower as an Embedded Software Engineer. I also spent some time later as an independent software consultant. In 2008 I returned to NAU and enrolled in the M.S. Engineering program as a Science Foundation Arizona Graduate Research Fellow. After finishing the program in 2011, I was brought on as an instructor to teach CS112, CS249, CS301, and various sections of CS126L and CS136L.

With interest (and enrollment) in computer science on the rise, it is an exciting time to be an educator for the department. Students are excited about the material presented in class and eager to show off their work to classmates. Also, with the small class sizes and the pair-programming structure of the introductory labs, students have been quick to collaborate and learn together. It is that enthusiasm and learning experience that makes an NAU education unique. It has also made teaching an enjoyable experience.

OUTSTANDING FACULTY WHO CARE

The Computer Science faculty has varying professional specialties, but all are united by a strong commitment to teaching. Among the special interests of the faculty are Groupware Systems and Intelligent Interfaces (Doerry), Graphics and Computational Geometry (Palmer), Distributed Systems and Web Technologies (Otte), Software Engineering and Robotics (Georgas), Bioinformatics (Caporaso), and introduction to software development and user interfaces (Jacobs).

This academic year, we have also invited some new, additional instructors to help out: Darryl Brown, Jeff Gordon, and Patrick Kelley.

Please come see us anytime, make an appointment, ask questions, try our interesting classes, sponsor a project, make a donation, join NAU ACM, tell a friend. There are many ways to get involved. See more about our program at:

http://nau.edu/cefns/engineering/computer-science/