ACM@CMU is proud to present the first issue of ACM Communications. We are extremely excited and hope you find the articles put together by fellow students enriching and interesting.

But wait, what is ACM@CMU?
ACM@CMU is the Carnegie Mellon chapter of the Association of Computing Machinery. We are a nascent, dynamic organization and were officially chartered with ACM in the Spring semester.

And what do you guys do?
To keep it simple, ACM@CMU helps you, one of the brightest students in the world, realize your potential. What makes CMU students stand out is not their academic backgrounds, but their ability to think, innovate, and apply what they learn. ACM@CMU is here to provide a platform for students to realize their potential and understand the scope of what can be achieved. We do this by organizing persistent, activities that enable students to work directly with leading industry members like Microsoft and Facebook, connect with professors, learn about cutting edge research positions, and understand the scope of various technologies and work on large scale projects.

ACM Communications? Why should I read it?
ACM Communications is meant to increase awareness about the latest developments in technology, both within and outside CMU in a way that it can enable students to capitalize on the opportunities available. We do this through features about the latest news, innovations, research, and how you can apply each of them. We help you connect with faculty and industry through guest articles and interviews. Most importantly, ACM Communications is personalized to CMU and maintained entirely by students so that you get to read about the best available opportunities and get a unique perspective on each of them.

So how can I get involved?
You can get involved by:
1. Being a part of the editorial board of ACM Communications. This can involve writing, editing or managing articles, depending on what your interests are.
2. Subscribe to ACM Communications and receive the issue whenever it is released.
3. Be a part of ACM@CMU. You do not need to have any particular interests or experience. As long as you are passionate enough to contribute by coming up with ideas and implementing them, or helping implement ideas, we have a spot for you.

To get involved, email us at acm-officers@lists.andrew.cmu.edu. For further information about ACM@CMU, visit our website at http://cmu-acm.org
We hope you enjoy ACM Communications!
ATTENTION facebook USERS

who owns your computer, right now?

Shashank Pradhan

Today it’s a norm for everyone to be a member on social networking sites like Facebook, Twitter, and MySpace regardless of whether they are 15 or 70 years old. This trend has been realized by hackers and they are targeting their energies to exploit this mass population of potential victims. There have been multiple attacks in the recent past on common networking sites like Facebook and Twitter. Just a simple mouse click could lead your computer straight into the waiting hands of a hacker or a cyber criminal!

Most of you would remember the koobface virus that hit Facebook last year. This virus prompted users to download software required to view a certain video. After downloading the “Flash upgrade” the users’ computers would all be infested with Botnets and spyware depending on the invariant of the virus.

A Botnet is basically a group of computers when infected with malicious code, turn into zombies or drones, and is controlled by a Botnet commander. Botnets are responsible for sending out 88% of spam in the world. These ‘bots’ are not only hurting you, but your friends and families as well. If your computer is part of a botnet, there is a good probability that when you post a link on a Wall or chat on Facebook it could lead to their computers being infected and become part of a Botnet.

Few months ago, Twitter servers were down all over the World. It was identified that the entire incident was caused due to a denial of service (DOS) attack.

A denial of service attack is basically an attempt to make a computer resource unavailable to its intended users using Botnets. The barrage of connection requests by Botnets overwhelms the sites, making it so that legitimate Web traffic can’t get through.

Researchers have even proved that a Facebook app can be a source of malware if written the ‘right’ way. There is a huge privacy risk involved with the site’s inadequate restrictions to third-party Facebook app developers from accessing users’ personal information. If I want to know “Which Archie Comics character” I am, the application should not have to and should not be allowed to snoop in my inbox for the fictitious answer that I get!

In today’s age, technology is something we cannot afford not to use regardless of how many threats or malware are out there. I am in no way dissuading anyone against using Facebook or Twitter or other similar networking sites. I am a big fan myself but great precaution should be taken while “Facebooking” or “Twittering”.

Some basic tips that could save you from being inducted into a Botnet are:

1. Change your passwords frequently
2. Be wary whilst clicking on random links
3. Finally and most importantly, do not believe any message that says you have won anything or there is a giveaway of any good.

Quoting the famous economist Milton Friedman, “There is no such thing as a free lunch.”
bureaucracies of the JDK

William Ng

Most of us either have, or will be exposed to programming in Java sometime or the other. Java is a friendly tool with an extensive pre-compiled optimized library. However, even Java can be a subject of confusion. I would like to provide some things that people should keep in mind when programming in Java.

Take for instance the modulus operator (%). Intuitively, one may think that it is going to return the same answer as the mod operation in mathematics. Unfortunately, in Java, the % operator returns an answer with the sign of the left operand whereas in mathematics the mod operator returns an answer with the sign of the right operand. For example: -5 % 3 = -2 whereas -5 mod 3 = 2.

Here is an example of seemingly correct code:

```java
    int addTop3(PriorityQueue<Integer> pq){
        PriorityQueue<Integer> temp = pq;
        return (temp.poll() + temp.poll() + temp.poll());
    }
```

Note that pq and temp are the same object. By calling poll() on temp, the original object has been mutated and that is clearly not the intention. Whenever you call a mutator method, ask yourself if intend to mutate an object. If you aren’t, then you better retract your call or clone the object if possible. It is also a good idea to separate methods in a class into mutators and accessors because it may not be explicit just by the name which one it may be.

Lastly, I would like to discuss some facts about Java inheritance and interfaces. Here are the rules:

1. Abstract classes and classes may extend from only 1 parent.
2. Abstract classes and classes may implement multiple interfaces.
3. Interfaces cannot implement other interfaces. They extend 1 or more interfaces.
4. When you call a method, polymorphism guarantees that it is called from the lowest level object even if you cast to a parent class.

A problem with these rules is multiple inheritances. It is quite awkward to get around this restriction. Another problem is implementing multiple interfaces with the same signature. There is no elegant way to solve this problem. The best you can do is to provide functionality with slightly similar signatures.

To sum up, in addition to efficiency, organization, abstraction, correctness, and other core features of programming, it is also important to keep in mind the minor details that can deter you a long way.
Can you give us a brief introduction about yourself, your life at and before CMU?
I (Gregory Kesden) have been at Carnegie Mellon for ten years. I teach systems courses, first year courses, and some courses in between. I also teach in the MSIT/eBiz program and direct our undergraduate laboratories. Before coming to Carnegie Mellon I was briefly on the faculty at Clemson University. And, before that I held jobs doing software development, AI research, and work on a DoD portal.

What are your interests for research and work outside courses?
I am primarily a teacher. Outside of teaching, I enjoy coaching the programming team, helping support great colleagues, and of course hanging out with students and helping them out in any way I can. I also enjoy hacking and do a fair amount of recreational programming.

What advice would you give to students in terms of expanding learning beyond coursework?
Love what you do -- and immerse yourself in it. If you have a 9-5 job, you lose 8 hours a day. If you have a lifestyle you love at an exciting place, you’ll accomplish a lot more, feel great about it, and have a tremendously great time.

We know you are greatly involved in ISR (Internet Suspend Resume). Can you tell us a bit about that?
We’re looking to it as a solution for a more agile undergraduate and outreach lab infrastructure. Instead of building a “Once configuration fits most” infrastructure, we want to be able to deliver a custom platform for every user community and deliver it to them wherever they are -- at home, on campus, or around the world. We don’t want users wasting time worrying about configuration, whether it is the configuration of the system, of the desktop, or their present work-in-progress. We just want them to be able to sit down, hit the “Go button” and be right where the left off -- regardless of where that was.

How can students contribute to the project?
OpenISR is a fairly young project. We’ve seen the proof of concept. And, there is a great framework. But, there remains a ton of work to be done in caching, performance optimization, policy development, the user interface, and the management tools. There might also be a lot that can be learned about user needs and user behaviors.

Is there anything else you would like to communicate to our readers?
I’m here to help. If I can ever be of service, please contact me. Drop by, email, or call. My office phone rolls to my cell phone 24x7.

For further information about ISR, visit http://isr.cmu.edu. If you would like to get involved with ISR, ACM is starting a cloud computing collaborative learning group. Email acm-officers@lists.andrew.cmu.edu for further details.


ants to protect your computer?

David Kim

Once again computer scientists are looking towards nature in search of answers. This time the subject is an ant and the field is computer security. Inspired by the swarm defense of ants, Computer Science professor Errin Fulp from Wake Forest University, working with graduate students Brian Williams and Wes Featherstun, developed a new type of computer network security software modeled after ant behaviors. Called “swarm intelligence,” the defense is meant to be active instead of the traditional static defense. “Ants can ramp up their defense rapidly, and then resume routine behavior quickly after an intruder has been stopped,” Fulp explains, “we were trying to achieve that same framework in a computer system.”

To avoid unnecessary residence within a machine, the ants are controlled by “sentinels” located within each machine, which are in turn controlled by “sergeants” monitored by humans. This hierarchy ensures that an optimized number of ants are deployed to each machine.

Utilizing ant behavior is only a small step towards smarter antivirus programs. By imitating and integrating natural habits that have been selectively chosen over millions of years, security experts now have new methods to help the digital world.

new security tactic that models nature

The main problem with current security software is that as more threats are realized and the definitions are updated, computer scans take longer and gobble up more resources. Though scans have been sped up through parallel processing, it pales in comparison to the ant swarm technology, which is faster by a thousand fold. Moreover, the high mobility of ant modeling will enable it to readily adapt to threats that are constant and always changing.

Like parallel processing, “swarm intelligence” divides up the work, not into 2 or 4 but thousands. Each ant acts as a separate searcher that crawls through the networks probing for a possible threat. As these ants move about they leave behind faint digital scent trails which attract other ants. When a possible threat is identified the ants are programmed to leave behind a stronger “scent,” leading to more ants being attracted to the site of infection and producing a swarm that tags possible threats for security experts to analyze. Human operators can then identify the threat and take proper action. Because of this, large networks with identical machines will receive the most benefit from the swarm intelligence.

There is a fire spreading throughout an apartment building. The security guard has noticed this and notified the fire department. He then calls 2 people and tells them to call two other people and get out after doing so. If each phone call takes 30 seconds and it takes 90 seconds to get out of the building, how long does it take to clear out the apartment if there are 125 residents living in the building (guard not included)? You may start measuring the time the moment the guard calls up someone (i.e. his time on the phone counts).

You can assume that:
1. Each resident live separately and that they are all present.
2. There are no duplicates in calls and that everyone calls someone else who hasn’t been called.
3. Finally a person may leave after calling only one person if everyone has been called.

The first three correct entries will win goodies from our company sponsors!