Joseph Elliott – SE Reflection

CR1. How do you think ISU has prepared you to:

a. **Design systems or processes?**
b. **Formulate and solve engineering problems?**
c. **Impact engineering solutions in a global / societal context?**

ISU has done various things to help prepare me for designing systems and processes. Most notably, my course Object Oriented Analysis. In this course, I've been learning about design with UML, CRC cards, and use cases. This class, along with a few others like SE 309, SE 329, and SE 339, have also helped in my ability to design.

In regards to solving problems, ISU has done a great job. Many classes I've taken revolve around problem solving, most notably CS 311, CS 327, and CS 430. These classes have forced me to think through my problems step by step – not rely on Stackoverflow.

Lastly, ISU has done a decent job at preparing me for global / societal engineering solutions. Software extends across the globe, and being able to identify solutions in these types of software is important. Now, I say that ISU has only done a "decent" job at preparing me for this because much of what I worked on here was individual-centric. Of course, I often worked with groups, but the projects I worked on weren't oriented to the global scheme. Everything was more or less "here's a tool you can use, now try it".

CR2. What things have you done at ISU to prepare you to:

a. **Work in groups?**
b. **Recognize contemporary issues?**
c. **Understand professional and ethical responsibilities?**

I've worked in groups throughout my entire college career. Whether the group work is for a gen-ed, programming course, or even extracuricular activities, I've almost always had a group to work with. For most courses, group work wasn't exactly allowed, but bouncing ideas off each other was. The cumulation of group work over the years has definitely added up.

Contemporary issues are hard to define as they are constantly evolving. ISU has provided courses (in the later years) that relate to modern problems, but not so many of them. Often times, I've found myself using out-dated software / hardware. For example, in CprE 288, we used old, usually broken robots, which made developing difficult. In other classes, we were often told to use much older versions of Java, Hadoop, etc. I understand the importance of legacy code, but at some point we need to learn the new tools.

This last part, understanding professional and ethical responsibilities, is a bit tough to answer. As far as being professional, ISU has done a great job. The Engineering Career Services has been the most helpful tool is getting hired, and has proved time and time again that it has the potential to connect you with corporations. However, the ethical responsibilities seems to have been forgotten. I know that Computer Science students are required to take a philosophy course about computer ethics, but us Software Engineering students are not. The only ethics-related material I've encountered was by a lecturer (a great man from the industry) who was just answering student questions.

CR3. In class projects and problem solving tasks, did you draw upon information, research or experiences beyond what was provided in class to successfully complete your work? Please state what resources you used here and how they helped you to complete work (e.g. library resources, specific professional journals, experts in field, other students).

I am always drawing upon other sources to help with class projects. Programming courses at ISU often revolve around the idea of a general topic. ISU forces the students to learn on their own to accomplish tasks related to these general topics. I feel like that's why students often call certain courses hard – they're not having their hand held through the course, so they fail. With that being said, though, ISU could do a bit better in guiding people to other sources / techniques for problem solving.

As far as resources I've used myself, I often consult experts in the field for help (when Stackoverflow doesn't help.) For example, I was having trouble making a simple backend API for a class project, so I emailed a coworker from an internship. He provided me with a great tool (Sinatra for Ruby) that made development very easy. Other than the experts, I often work with other students. Multiple brains is almost always better than a single brain. I've worked through countless problems with friends that I couldn't solve by myself.

CR4. How did learning activities outside of the classroom (required 124.5 credits) - such as Student Orgs, Career or Study Abroad Fairs, Undergraduate Research Experience (REU), or other university programs - help you to understand the importance of Lifelong Learning?

The Career Fairs and classes at ISU got me in contact with great people. These people all had something that I didn't know was common past college: the desire to learn new things. At my internship at Cerner, my team's architect provided me with a long list of books to help me in my software career. In my project management class, my instructor told us of how he "lost his way" in exploring software – an eye opening time for myself, because I didn't realize professionals lost their motivation after many years of coding.

CR5. Have you started to undertake any new learning to improve your ability to apply skills or knowledge to new problems and to develop confidence in taking risks? Please explain.

I'm always looking for something new to learn. I've recently taken an interest in neural networks, a potential tool to use in the future. I don't need to know about neural networks now, but maybe in the future, there will be a job that I can fight for. These potential jobs, I would identify them as risks worth taking. I also am always trying to further my knowledge of the Android platform so that, one day, I'll have the talent to try for a full time position as an Android developer.

CR6. In the context of the first four questions, if you were to do your undergraduate work again, what things would you change?

I would do everything the same exact way, but work harder. There's so much to learn, and so little time.