**CIS 104 Final Exam – Jon Weld**

1. What are the important parts of a loop? Why are they important? What are the two general categories of loops?

The most important parts of a loop are the initial statement that starts the loop, the statement in which the initial clause doesn’t work or doesn’t fit the input, the break or exit in the loop, the if else statements between the initial statement and the break, and the statement that is getting looped based off of the initial input and every input afterwards (ex. i > 5). The initial statement is important because it can determine what kind of loop you are looking to accomplish. The backup statement is important because the initial input may not always work so, like anything, you have a backup. The break or exit in the loop is probably one of the most important parts of a loop because without that, you have the potential to make the loop go on forever and it could overload whatever you are running on. The equation in the beginning statement is also very important because that is what the loop is based off of, and that is what determines how long the loop will last. The if else statements are important because they are the basis as to how your loop will perform in terms of what it calculates. The two general categories of loops are for loops and while loops. For loops are used when we are looking for if something is true, whereas a while loop is used in the case in which it loops while something is true. But even with their differences, they both use the if else statements for making the loops happen.

1. Describe how an application would use a database. Give an example of an application you use that has a database as a critical component.

An application, depending on what kind of application, uses a database to keep track of user information such as your first and last name, your account ID, your address, your phone number, etc. and depending on what kind of company application you are using it can contain other things such as your license plate number and your customer ID. The company also has to take into account how big of a database they will initially need because there are certain database applications (such as SQLite and Microsoft access) that can only hold so much information and that can be easily managed. One app that I use frequently, and that most of the world uses, is the Amazon app. They probably have one of the biggest databases in the world, if not the biggest, because people buy things all over the world all the time and they have to keep track of all of that info. They have information such as your name, phone number, email address, potentially credit card info if you allow it (but who knows if they still have it on file regardless of whether you save it or not), home address for shipping, and your past orders. But in terms of the primary key they use, it would most likely be a customer ID of some sort. In terms of what the name of their database application, I am not sure but it would have to be something that can hold billions of peoples’ information and their order info.

1. What does it mean to program with good style and structure? Why is this important?

To program with good style and good structure, you kind of have to compare it to the structure of an article or a newspaper. They have their chunks of information separated by white space and sub headings. Programming is very similar in that case. The way you keep a program looking nice is you have to divide your different sub processes with at least one line of white space between different chunks, you label the various chunks with their function such as: if you have a loop that calculates whether or not a value is greater than or equal to the given value in the loop, you would put a # on a line close to the chunk (for python at least) to label it with the fitting description (kind of like a sub heading in a newspaper). It is good to get into the habit of labeling your sub processes because if you are trying to show how your program works to someone who has never done any sort of code programming, you would be able to explain how it works to them a hell of a lot easier than if you were to not include the sub headings. That and if you were to share your code with a potential co-worker and they need to make adjustments to it, it would help them very much knowing that they had the different sub processes briefly described to them, so they know what to include and what to not include.

1. What is Git and why is it useful? Give specific examples of situations where you would want to use Git.

Git is a version control software that can be used for multiple kinds of management such as content and configuration management (techtarget.com). It is useful because as well as being a sort of coding language, it has a place for storing your coding projects called a repository that people can access (such as coworkers or fellow students) on their website called GitHub. It can be used for keeping track of major coding projects and what mini coding projects make up the big coding project. You can also add collaborators to whatever project you choose so they can work on the project while you are away, which really utilizes the time management aspect of a project. You can also use it for storing databases as well (which I have done with SQLite databases). But you have to be careful about who you add to your repositories and what permissions you give them for the repository because in the past there was at least one data breach on GitHub and it is not nice having to realize your coding project that you have been working on is all of a sudden nowhere to be found (that is why you should also get into the habit of making regular backups for your projects, big or small, but preferably the big ones). So, unless you plan on collaborating the project, you should always keep your repository or repositories private. But overall, Git and GitHub are very useful for keeping all of your projects in one place, may not be the most secure, but nothing is ever truly secure.