

Hunter McMillen  
ISAT 252  
Morgan Benton  
4/22/10

1

## My ISAT 252

When I first came into this class I wasn't really sure what to expect, I just thought it would be another programming class where I take notes on the ins and outs of Visual Basic, get programming assignments and take tests every so often. Turns out, that isn't what happened, and I am very happy about that.

The course was interesting to me, despite already knowing many of the object oriented programming concepts. I feel as though I learned as much about visual basic in this course as I would have in a regular programming course but without all of the boring lectures and programming assignments unrelated to topics we are studied. In particular, the section we just learned about knowledge based and expert systems was really interesting, until then I had never heard of a knowledge based or expert system and it was cool to learn what they are and their applications in the software world. My most important accomplishment, to me, was created a full binary subtracter in the Mmlogic program.

That being said, because of my good attendance, effort put into the course, good grades on everything we have turned in, and my belief that I have successfully fulfilled all of the course objective laid out in the syllabus, I believe that I deserve an 'A'.

Here is a listing of the course objectives and what I've done to fulfill them:

**1. Demonstrate a basic familiarity with computer hardware and explain the function of the main components of a computer**

I fulfill this objective with my answer to the first test question on test #1.

**2. Recognize when a problem in science, business, or engineering is amenable to a software solution**

At any point in time, if software can solve a problem easier than a previous way, software should be used.

**3. Analyze such problems and generate plans for implementing software solutions including elements such as class diagrams, flow charts, and pseudocode**

Mark and I used class diagrams and screen drawings to model our battleship program before we started coding.

**4. Implement relatively simple software applications with**

Hunter McMillen  
ISAT 252  
Morgan Benton  
4/22/10

2

**graphical user interfaces using a modern programming language such as Visual Basic .Net**

For Lab 6, we were assigned a problem by another team, and I used an array implementation to store student informations and used buttons and labels to retrieve that information.

5. **Employ procedural programming constructs effectively: variables and constants, conditional expressions, flow control, arrays, modularity with sub-procedures and functions, and input/output with text files**

Also in Lab 6, I used array to store student information and several functions to return that data.

6. **Employ event-driven programming constructs effectively: built-in and custom event-handling procedures, catching and handling exceptions**

In our Battleship program, Mark and I used event handlers to determine when a missile shot at a ship was a hit or a miss.

7. **Employ simple object-oriented programming constructs: distinguish between objects and classes, identify when classes would be useful in a program, implement a basic class, and use that class in a program**

Also in our Battleship program, we pass data from class to class and have many classes, if we didn't have all of those battleship classes our program would look very procedural.

8. **Use a range of debugging techniques such as use of online and offline reference materials, search strategies, online forums, and integrated debugging tools**

I used visual studio's debug tool as well as hand traces solve any exceptions or errors caused by programs I have written this semester.

9. **Document code appropriately to increase ease of maintenance and understanding**

I put good comments into my code to ensure that anyone reading it can understand what is going on, this is a concept that has been drilled into me for awhile.

10. **Read and understand code that others have written and be able to predict the results of running that code and/or re-use/incorporate others' code when appropriate**

Mark and I worked collaboratively on the Battleship program so when I

Hunter McMillen  
ISAT 252  
Morgan Benton  
4/22/10

3

was adding functionality to the program I had to be able to understand what he had already done to the program.

**11. Describe what a knowledge-based system is and the types of problems and solutions to which it is best suited**

A knowledge based system is a system that is built to reflect the knowledge of someone who is an expert on a topic. They are best used to make choices between different options, whether the options are where to eat, what car to buy, what medicine to take, etc..

**12. Describe the architecture of a typical expert/knowledge-based system, describe the methodology used to develop expert/knowledge-based systems, and use the methodology to develop an expert system**

A knowledge based system is made up of variables, values, and goals. The variables are the conditions of the decision and the goals are the result. I used goals, values, and variables to develop a small KBS system for the lab due on Friday.

**13. Use 3 forms of knowledge representation: pseudo code rules, decision trees and rules in a program/shell rule base**

For the lab due on Friday, I used decision trees and rules to map out the KBS system on the Corvid shell.

**14. Extract production rules from a narrative, identify key variables, and construct correct decision trees in an applied problem-solving scenario**

I did this in our second KBS homework assignment. KBS HW #2

**15. Evaluate a given problem within its social context and identify the most appropriate paradigm within which to develop a software solution**

I only paradigm that I have studied to any extent is the object-oriented one, so I feel that for most problems I try to solve I will be using oop, procedural programming was very popular, but the capabilities of an object and reusable code outweigh it.