David Medina
Jordan Miller

Design Reflection

1. One major change was the deletion of the block superclass and the addition of the statement data member to the Ifs superclass. We felt that it would be simpler to pass a SectionToken of type block to a data member to the corresponding Ifs which would in turn make it easier to implement the execution of the blocks as the they would be apart of its corresponding Ifs. Also includes the creation of a getComplete member function in the Ifs superclass in order for processing the associated blocks.

2. Another change was the removal of the run functions with parameters int and string in favor of a run function with no parameters. Initially we though we might do the run function with an index or a string to process a passage and the following passage but it ended up easier to implement just one run function that is called once and has the next passages to be processed be determined within the function.

3. Another change was the addition of more data members into the interpreter class such as a vector of links, a hitGoto, and a hitLink.The vector of inks, hitGoto, and hitLink were initially inside the run function of the interpreter but we needed the execute functions of our section superclasses to edit those values so we implemented them into the interpreterclass. These data members were mostly used to determine if the the next passage to be processed with links and if the story was at its end if hitLink and hitGoto are 0.

4 The last major change was the creation of a position variable with corresponding mutator and accessor member functions. We found it easier to implement the position variable(pos) which determines what the next passage to be executed is based on links and gotos updating the pos which replaced some of the member functions of the interpreter like findIndex, updateVar, and getVal.

5. Minor Changes

- Section execute functions now take an interpreter class as a parameter in order to update and access essential data members of the interpreter.
- Replaced our planned story tokenizer from Project part 1 for the sample story tokenizer provided as we felt it would make implementing the IF interpreter easier as the sample wouldn’t have the bugs our story tokenizer had.