Comp 110 - Research Journal

This paper will be focusing on finding a link between various articles on computing and how humans interact with them. By the end of this paper I will have analysed the different papers and given my opinion on them whilst also finding connections between them all to make an appropriate conclusion.

Paper 1

Clare Horsman asks the question: ‘When does a physical system compute?’.
She explains how organic computers are untrustworthy as they are more likely to make molecular mistakes than digital systems which already make errors.
I would argue that this is irrelevant in gaming, particularly Artificial Intelligence. Having errors present in game which causes the A.I to malfunction and behave incorrectly due to mistakes on a molecular level within the system works to the advantage of the developer.
Intelligence as we know it, is adapting to a situation and learning from it, so from the perspective of the player, it will seem as if the system is learning from its mistake.

Horsman, Clare, et al. "When does a physical system compute?." Proc. R. Soc. A. Vol. 470. No. 2169. The Royal Society, 2014.

Paper 2

The stages of group planning have had several different approaches made toward them. Notable researchers such as: Lacoursiere(1974), Spitz and Sadock(1973)
all concluded, with a few anomalies and disagreements as to how many stages there are, that the cycle is “Anxiety, Anger/dissatisfaction, productivity, termination”.
I believe, however, that this is not a wholly credible source as the studies were primarily conducted on white females in their second year of nursing (with the exception of one male).

Tuckman, Bruce W., and Mary Ann C. Jensen. "Stages of small-group development revisited." Group & Organization Management 2.4 (1977): 419-427.

Paper 3

“Experimental Investigations of the Utility of Detailed Flowcharts in Programming”
This paper described the research which a group of people collaborated on to observe the effects of flowcharts in terms of productivity for the team using them. The researchers pitted groups with flowcharts against groups with simple programs.
The study showed that the group without the flowchart would sometimes be noted to outperform the other group in terms of efficiency. Gerald M. Weinberg said that he found “no evidence that the original coding plus flow diagrams is any easier to understand than the original coding itself except to the original programmer.”.

Shneiderman, Ben, et al. "Experimental investigations of the utility of detailed flowcharts in programming." Communications of the ACM 20.6 (1977): 373-381.

Paper 4

Edsger Dijkstra proposed that all ‘go to statements’ be removed from higher level programming languages due to people over complicating the code making it harder for others to read and interpret. I feel the validity of this paper is questionable regardless as the date this paper was written was 1977 which is unlikely to have had similar debugging processes as modern day systems have. Therefore perhaps the removal of the ‘go to statement’ from most higher level languages, which is what he proposed, could be reversed to see an increase in productivity.

Gilbert, Elmer G., Daniel W. Johnson, and S. Sathiya Keerthi. "A fast procedure for computing the distance between complex objects in three-dimensional space." IEEE Journal on Robotics and Automation 4.2 (1988): 193-203.

These four papers are all looking at the best ways to maximise productivity. The four papers all approach the topic of productivity from slightly different angles and perspectives. Horsman (2014) approaches productivity specific to computing, arguing that digital systems are likely to lead to more productivity than organic systems. The papers by Shneiderman *et al*. (1977) and Gilbert, Johnson & Keerthi (1988), also look at productivity from the perspective of computing, however, the focus is slightly different. Shneiderman *et al*. (1977) focus specifically on productivity relating to flowcharts, whereas Gilbert, Johnson & Keerthi (1988) focus upon productivity relating to higher level programming languages (and in particular 'go to statements'). The paper by Tuckman & Jensen (1977) also looks at productivity, but it is not specific to computing, they are simply looking at the stages of group planning overall. By analysing the different stages of group planning in this way, you can identify the most productive stages for groups, and thus know when the most appropriate time for work to be undertaken is, in order to increase productivity.