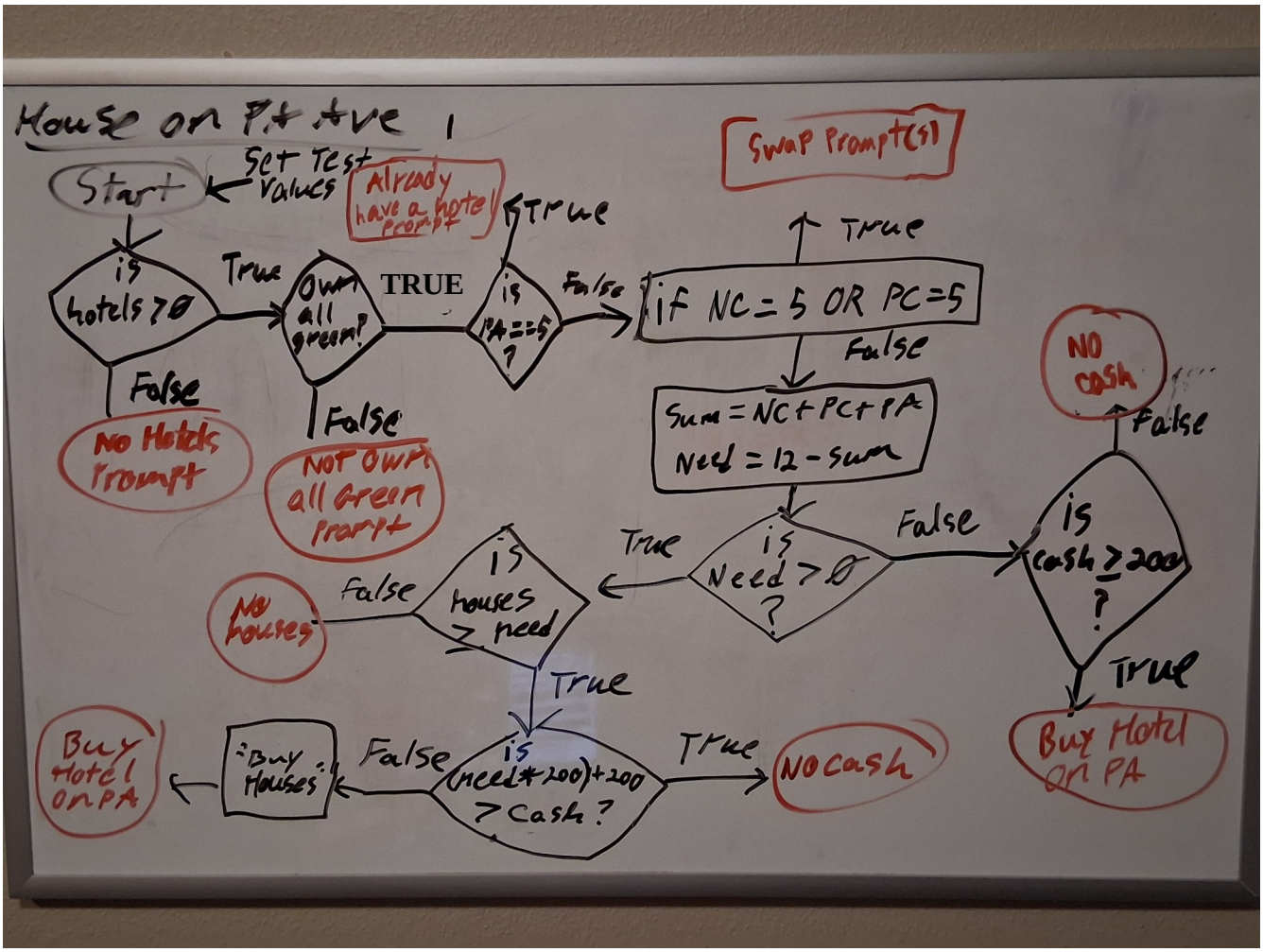
Patrick T. Edgett

CSE130 – Algorithm Design

Lab 03: Monopoly Design – Buy House on PA Avenue

1/31/26



The above is my second draft of my Monopoly Program that I will implement over top of my prior designs.

The Flow chart does account for the Multiple outputs for buying houses before buying a hotel, but I ran out of space on my white board. I can provide a better version at a later date if requested, but I think this should be legible enough to convey what I'm doing.

Just in case, broadly, the program will start by initializing all the parameters at once, then check if there is any Hotels available. Since we'd only need 1 if any, I check if it's Greater than 0. It then checks if you own all green properties, if True it will check if there is already a hotel on PA before checking if there is a hotel on any other property. Hotels will be represented by the integer 5 for simplicity.
Once the check for hotels has been completed, we then check to see how many houses we have. Since the required number of houses for a hotel is 12, we simply take the sum of all the houses on the properties and subtract it from 12 to get our needed houses. Once we've determined if Need is Greater-Than 0, which would mean we would need to buy houses, we will either check to make sure we have the minimum price for a hotel, or move to check if the bank has enough houses. If they do, we check to see if we have enough cash to buy all the houses and hotels. Houses and hotels cost 200 a piece, so we can find the cost of the houses and then add 200 to compare to our cash input. If it passes, the "Buy Houses" block will run before sending the output stating you've bought a Hotel on PA Avenue.

The process/function "Buy Houses" will check each property variable, if the property equals less-than 4 it will output: "Put {4 – x} house(s) on {property name}", otherwise it will not print anything.