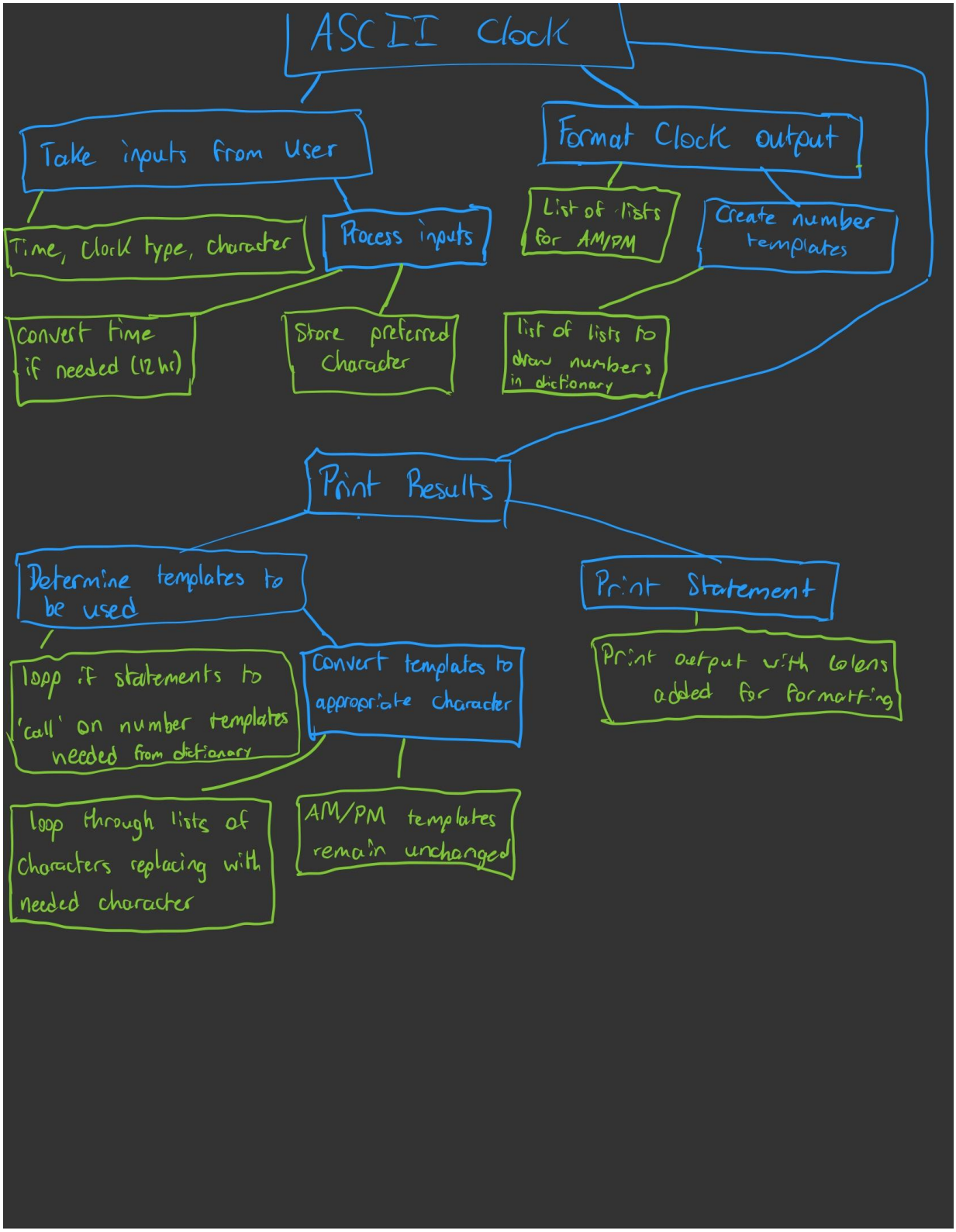
Rivan Adhikari
JT Swiderski
Patrick Simon
Cody Wu



Variables:
Mil: A list of list of strings that displays AM or PM depending of if the user had specified 24 hour time or 12 hour time.

PM: A list of list of strings that displays PM.
AM: A list of list of strings that displays AM.
Numbers 0-9: A list of list of strings that displays numbers 1-9. Shaped by the corresponding number.
Colon: A list of list of strings that contains colons for the clock output.
Time: User input variable that holds time.
Type: User input variable that holds 12 hour time or 24 hour time
Char: User input variable that holds the character the user specifies to make up the output clock.
Valid: A list of valid characters for the char input.
Hour: The hours portion of the time entered by the user. May be converted to 12 hour time if user enters the 12 hour type.
Minute: The minutes portion of the time entered by the user.
Output1-5: These variables will store each row that will make up our final output

Test Cases:

- Middle cases
    - 3:00, 12, $
    - 8:00, 24,
    - 17:00, 12, *
- Edge cases
    - 12:00, 12, f → p
    - 0:00, 12, a

Describe the difficulty your team had when combining the code at the end. How could your team have specified the design more clearly at the beginning?

- Our team had some trouble combining the code at the end. Between logic and syntax problems, we had to fix a good amount of problems. Sometimes our code would run individually but not within the whole program. This meant that we had to edit code so that everything would flow efficiently. It was also important to test the entire program as a whole with the test cases that we came up with beforehand. We think that if we were more specific at the beginning and had more in depth pseudocode then it would have given us a better idea of what exactly we needed to do.

Describe any benefits and drawbacks you saw by dividing the coding like this. Can you see reasons why this might be a good idea? How about a bad idea?

- This can be beneficial as it helps to combine four minds compared to one. It helps us cover our individual shortcomings. However, this method does not work if one group member is not doing their part. It also can make us susceptible to groupthink. We believe that in a good group that collaborates well by bouncing ideas off each other and has a real drive to do well and succeed, group coding is extremely effective. If our group was conflicting and not amicable, then it would be difficult for us to get stuff done, and it would just be better to code individually.