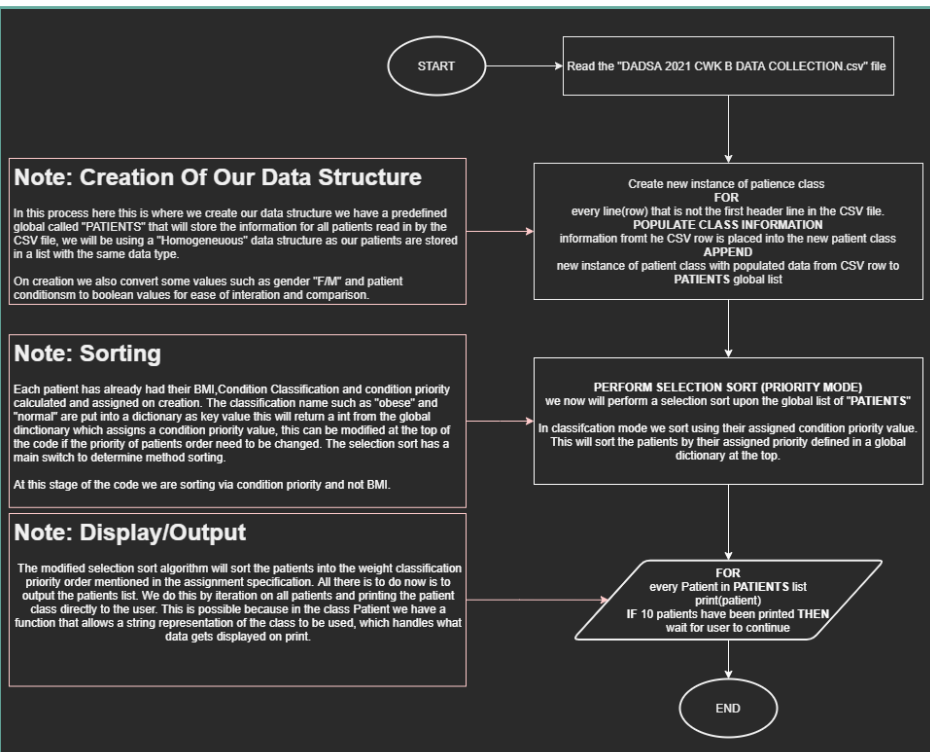# FLOWCHART DESIGN DIAGRAM FOR TASK 1
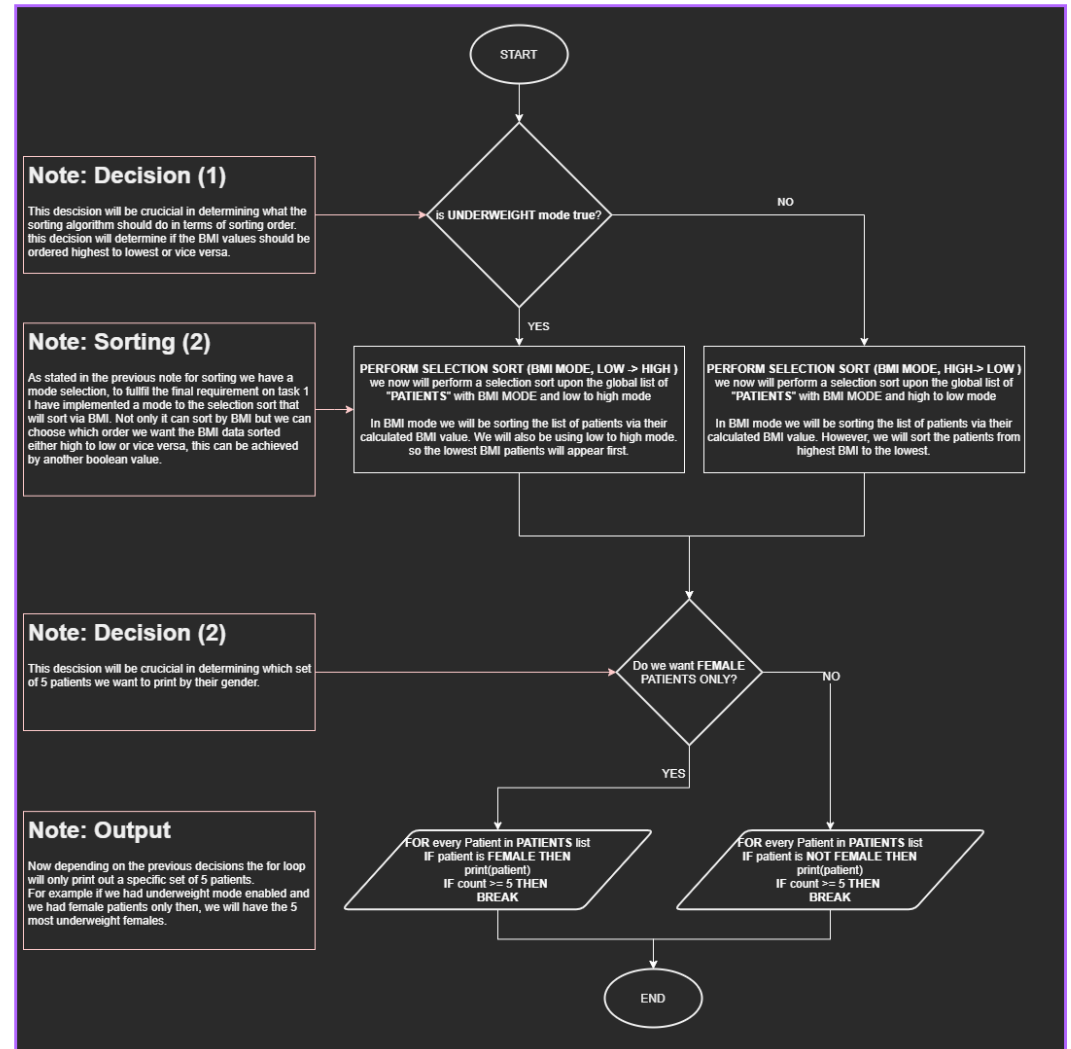
Flowchart for Task 1

The flowchart below describes the process of the solution.py for task 1. It will only cover the read/construction of data and output process for printing patients by their weight classifcation order as mentioned in the spec.

START → Read the "DADSA 2021 CWK B DATA COLLECTION.csv" file

Note: Creation Of Our Data Structure

In this process here this is where we create our data structure we have a predefined global called "PATIENTS" that will store the information for all patients read in by the CSV file, we will be using a "Homogeneuous" data structure as our patients are stored in a list with the same data type.

On creation we also convert some values such as gender "F/M" and patient conditionsm to boolean values for ease of interation and comparison.

Create new instance of patience class
FOR
every line(row) that is not the first header line in the CSV file.
POPULATE CLASS INFORMATION
information fromt he CSV row is placed into the new patient class
APPEND
new instance of patient class with populated data from CSV row to
PATIENTS global list

Note: Sorting

Each patient has already had their BMI,Condition Classification and condition priority calculated and assigned on creation. The classification name such as "obese" and "normal" are put into a dictionary as key value this will return a int from the global dinctionary which assigns a condition priority value, this can be modified at the top of the code if the priority of patients order need to be changed. The selection sort has a main switch to determine method sorting.

At this stage of the code we are sorting via condition priority and not BMI.

PERFORM SELECTION SORT (PRIORITY MODE)
we now will perform a selection sort upon the global list of "PATIENTS"

In classifcation mode we sort using their assigned condition priority value. This will sort the patients by their assigned priority defined in a global dictionary at the top.

Note: Display/Output

The modified selection sort algorithm will sort the patients into the weight classification priority order mentioned in the assignment specification. All there is to do now is to output the patients list. We do this by iteration on all patients and printing the patient class directly to the user. This is possible because in the class Patient we have a function that allows a string representation of the class to be used, which handles what data gets displayed on print.

FOR
every Patient in PATIENTS list
print(patient)
IF 10 patients have been printed THEN
wait for user to continue

END

Flowchart for Task 1

This flow chart covers the process for the last requirement for task 1, I seperated from the other main flow chart for simplicity and readability.

START

Note: Decision (1)

This descision will be crucial in determining what the sorting algorithm should do in terms of sorting order. this decision will determine if the BMI values should be ordered highest to lowest or vice versa.

is UNDERWEIGHT mode true? — NO / YES

Note: Sorting (2)

As stated in the previous note for sorting we have a mode selection, to fullfil the final requirement on task 1 I have implemented a mode to the selection sort that will sort via BMI. Not only it can sort by BMI but we can choose which order we want the BMI data sorted either high to low or vice versa, this can be achieved by another boolean value.

PERFORM SELECTION SORT (BMI MODE, LOW -> HIGH)
we now will perform a selection sort upon the global list of "PATIENTS" with BMI MODE and low to high mode

In BMI mode we will be sorting the list of patients via their calculated BMI value. We will also be using low to high mode. so the lowest BMI patients will appear first.

PERFORM SELECTION SORT (BMI MODE, HIGH> LOW)
we now will perform a selection sort upon the global list of "PATIENTS" with BMI MODE and high to low mode

In BMI mode we will be sorting the list of patients via their calculated BMI value. However, we will sort the patients from highest BMI to the lowest.

Note: Decision (2)

This descision will be crucial in determining which set of 5 patients we want to print by their gender.

Do we want FEMALE PATIENTS ONLY? — YES / NO

Note: Output

Now depending on the previous decisions the for loop will only print out a specific set of 5 patients. For example if we had underweight mode enabled and we had female patients only then, we will have the 5 most underweight females.

FOR every Patient in PATIENTS list
IF patient is FEMALE THEN
print(patient)
IF count >= 5 THEN
BREAK

FOR every Patient in PATIENTS list
IF patient is NOT FEMALE THEN
print(patient)
IF count >= 5 THEN
BREAK

END

In the flowchart diagram above you can see my design for task 1. In this example I have described and outlined the processes of how my python script will handle and sort the patient information data and from the CSV file.