Our algorithm takes the aptitude scores for a given number of students and sorts them into one of the four Hogwarts houses in a way that maximizes “house happiness”. We chose to define house happiness as the average house-specific score of all of the students sorted into a house. For example, if Gryffindor had five students, with Gryffindor scores (each student has an aptitude score for each house) of 9, 7.5, 1, 8, and 3, Gryffindor’s house happiness would be 5.7. We compare this actual happiness to the maximum possible house happiness, which we calculate by taking the average of the top quartile aptitude scores for each house. In general, this ideal will never be achieved for all four houses, as a student who ranks very highly in more than one house may contribute to more than one maximum house happiness, while a student that doesn’t rank highly in any of them will likely not be present in any top quartile; however, in the actual house happiness calculation, each student will be counted only and exactly once.

In order to sort the students, our algorithm first places each student into their top choice house. If none of the houses are over capacity (as defined by one quarter of the total incoming students, plus or minus one to account for class sizes that are not evenly divisible by four), then house happiness has been considered maximized and the rosters are finalized.

However, in the (highly likely) case that students are not evenly distributed amongst the four houses, our algorithm seeks to relocate students in a manner that will least disrupt the house happiness. This is achieved by looking for students with the least difference between their first and second aptitudes, where these least difference students are then moved to their second choice house. The algorithm then checks house sizes again, and will repeat the relocation process as many times as is necessary until houses are even sized.