Theoretical WC:

1. First we place all the indices into groups of 4. This requires n/4 comparisons. If n is not divisible by 4, then an “odd” group is created that is dealt with at the end. For this WC scenario, the odd’s existence doesn’t matter, as its weight is worthless for reasons explained later. For the groups of 4, each of them are further divided into one of 3 groups: a group with a weight of 0 (which is thrown out), a group with a weight of 2, and a group of a weight of 4.
2. At this stage, the WC would be if the entire set of groups were groups with a weight of 2. This is because each 2-group requires an additional comparison in relation to a 4-group to determine the majority indices within that group. Additionally, for the first step, as there is no group with a majority for reference, the first group requires 2 additional comparisons. Finally, as this is WC, no groups get factored out at this stage.
3. In this WC scenario, all the remaining groups are compared 4 groups at a time, with 1/4 of the groups being pushed for each round, with possibly one case of varying weights occurring at the ends of each iteration depending on the n value. The difference is statistically insignificant however for large values of n. Each of these push up decisions takes 2 comparisons per 4 groups. This continues until only 1 group is left.
4. Finally, the final group’s majority index is returned. Here is where the “odd” group would normally come into play, but as the weight of final group is higher than 3 for all values of n>=16, the “odd” group can be ignored in most cases.

WC = (n/4) + (n/4 + 1) + $\sum_{i=1}^{log4\ n} (n/(4 * 4^i)) * 2$ + 0

Theoretical Expected WC:

For n = 20, the theoretical expected WC will be about:

(20/4) + (20/4 + 1) + $\sum_{i=1}^{log4\ 20} (00/(4 * 4^i)) * 2$ + 0

= 5 + 6 + 3.125

= 14.125

For n = 200, the theoretical expected WC will be about:

(200/4) + (200/4 + 1) + $\sum_{i=1}^{log4\ 200} (200/(4 * 4^i)) * 2$ + 0

= 50 + 51 + 32.8125

= 133.8125

For n = 2000, the theoretical expected WC will be about:

(2000/4) + (2000/4 + 1) + $\sum_{i=1}^{log4\ 2000} (2000/(4 * 4^i)) * 2$ + 0

= 500 + 501 + 333.0078

= 1334.0078

Theoretical AVG:

1. First we place all the indices into groups of 4. This requires n/4 comparisons. If n is not divisible by 4, then an “odd” group is created that is dealt with at the end. For the groups of 4, each of them are further divided into one of 3 groups: a group with a weight of 0 (which is thrown out), a group with a weight of 2, and a group of a weight of 4. For AVG, 50% will be 2-groups, 12.5% will be 4-groups, and 37.5% will be 0-groups.
2. Next, we discover the majority indices of all the 2-groups. Each of these discoveries will take 1 comparison, which in total is n/8 comparisons. Additionally, about 50% of those n/8 groups will be eliminated due to their majority bits appearing in the 1st 2 indices checked, which allows them to be eliminated in our formula. An proxominate ¼ of the remaining groups are also eliminated in this process due to them canceling each other’s weight.
3. In this AVG scenario, all the remaining groups are compared 4 groups at a time, with 1/4 of the groups being pushed for each round, with possibly one case of varying weights occurring at the ends of each iteration depending on the n value. The difference is statistically insignificant however for large values of n. Each of these push up decisions takes about 1.3125 comparisons per 4 groups, as ⅛ the time it takes 1, ½ the time it takes 2, and ⅜ the time it takes 1 and completely eliminates all 4 values from the tournament. This continues until only 1 group or no groups are left.
4. Finally, the final group’s majority index is returned or a 0 is returned, depending on the results of step 3. Here there is a chance 1 or 2 comparisons are needed, depending on whether an “odd” group exists and how many extra values it represents. We will say that on average, it adds 0.75 comparisons.

AVG = (n/4) + (n/8) + $\sum_{i=1}^{log4\ n} (n * 9/(128 * 4^i)) * 1.3125$ + 0.75

Observed WC and AVG + Inconsistencies:

When we ran our code for 10,000 loops, we resulted in:

n= 20, max= 14, avg= 8.64

n= 200, max= 98, avg= 81.88

n= 2000, max= 861, avg= 809.55

For smaller n like n = 20, there are 2^20 possible permutations. In our run of 10,000 loops, a worst case permutation occurred.

Compare that to n = 200 with 2^200 possible permutations, or furthermore n = 2000 with 2^2000 possible permutations, and it is highly unlikely that the particular worst case scenario would occur, thus explaining our large differences in this area.

For Average, if any inconsities occur in relation to our theoretical average, it is most likely because there are hidden mechanics at work that resulted from the solution’s complexity and thus end up slightly changing the results.