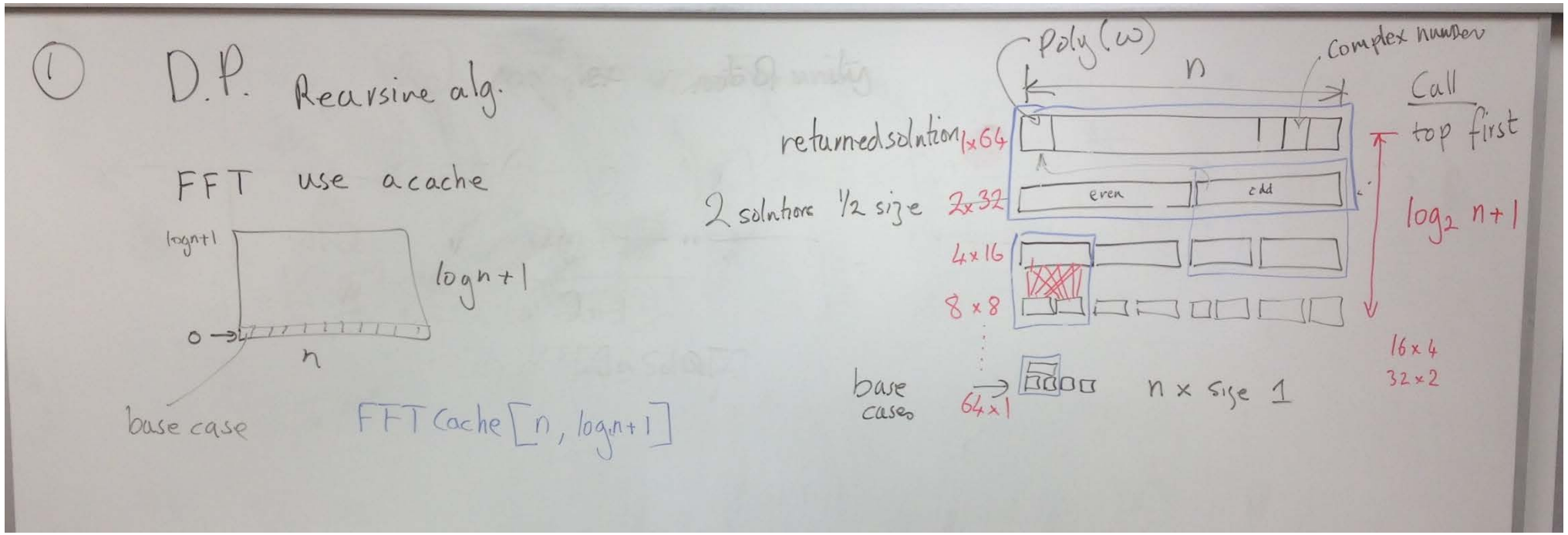
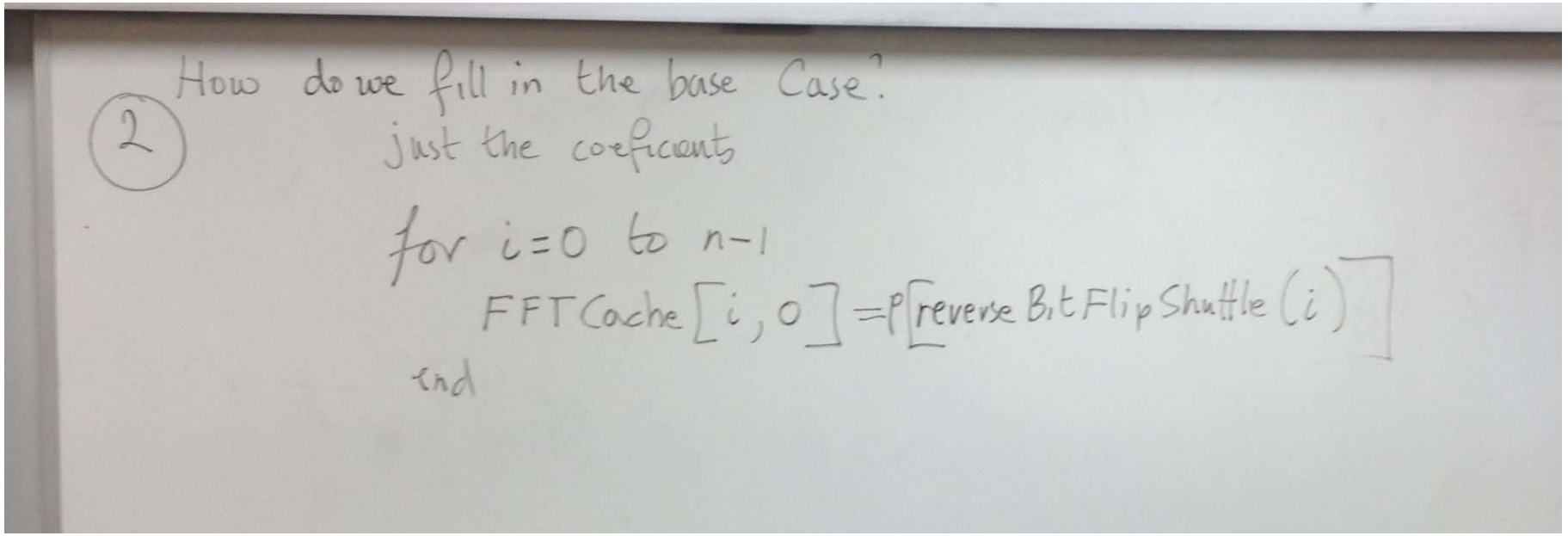
# cs5050

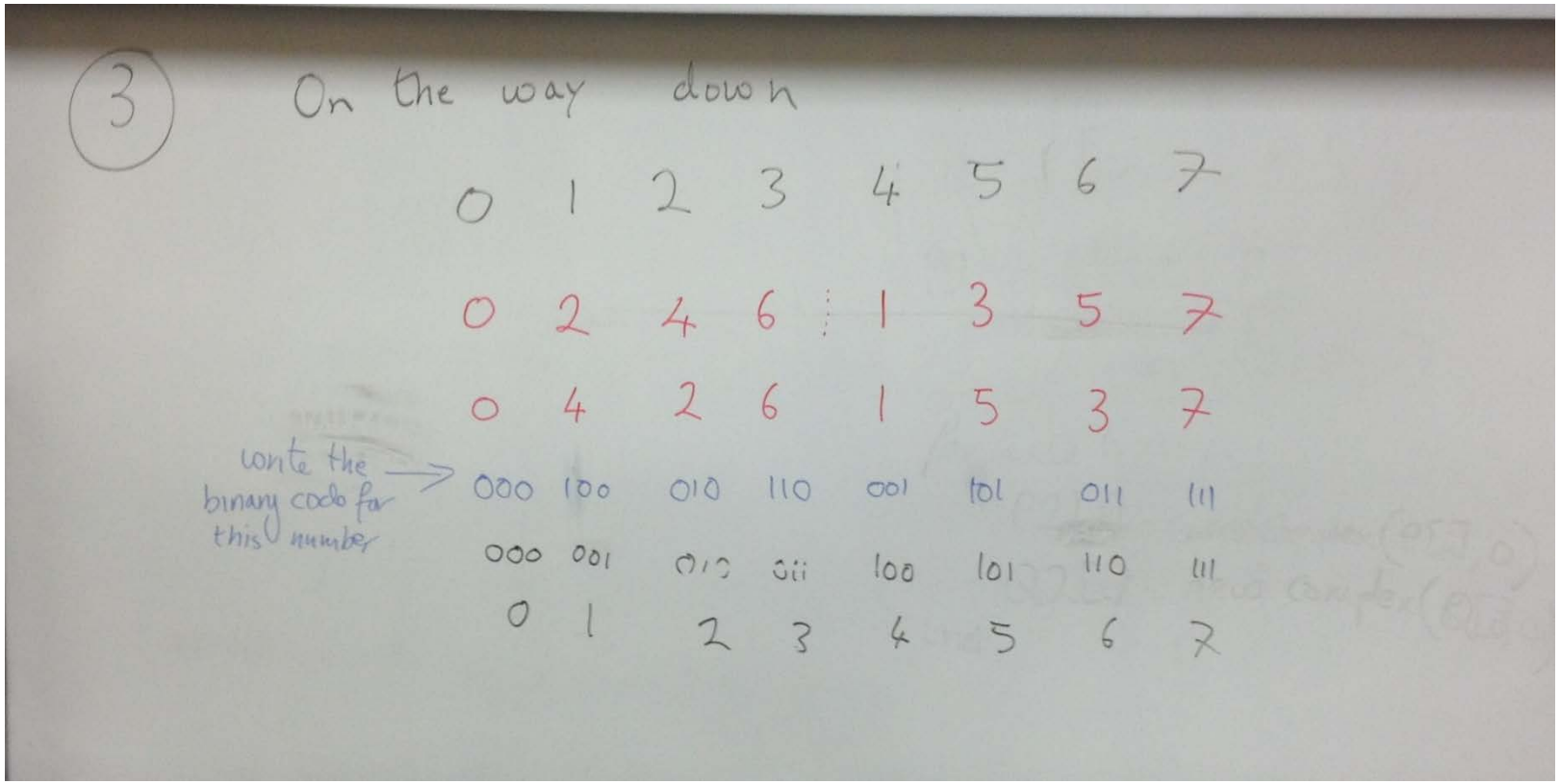
03 06 14



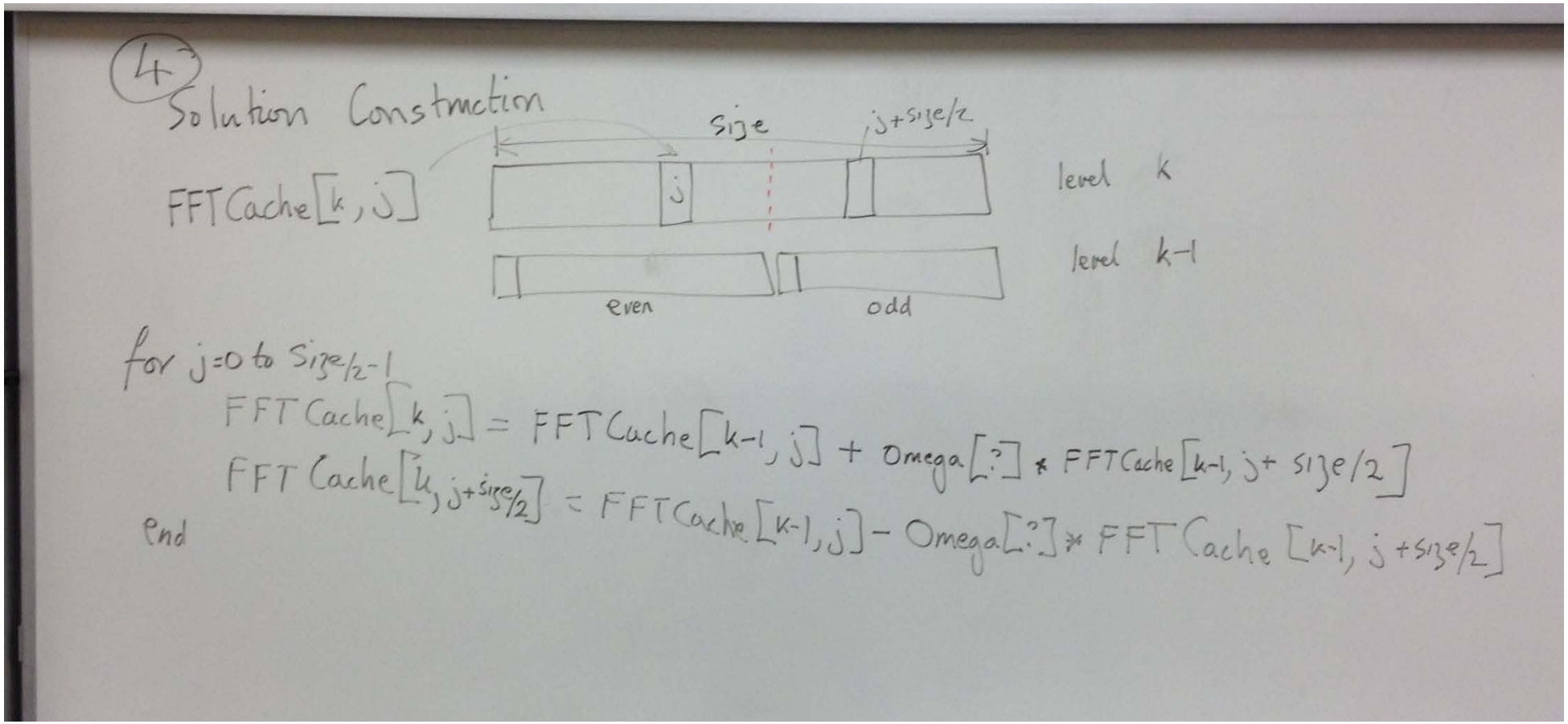

Cache contains all the solutions computed.
By analysis of the calling tree of solution/subsolutions we can determine there are n by $\log_2 n+1$ solution values computed by the recursive function.



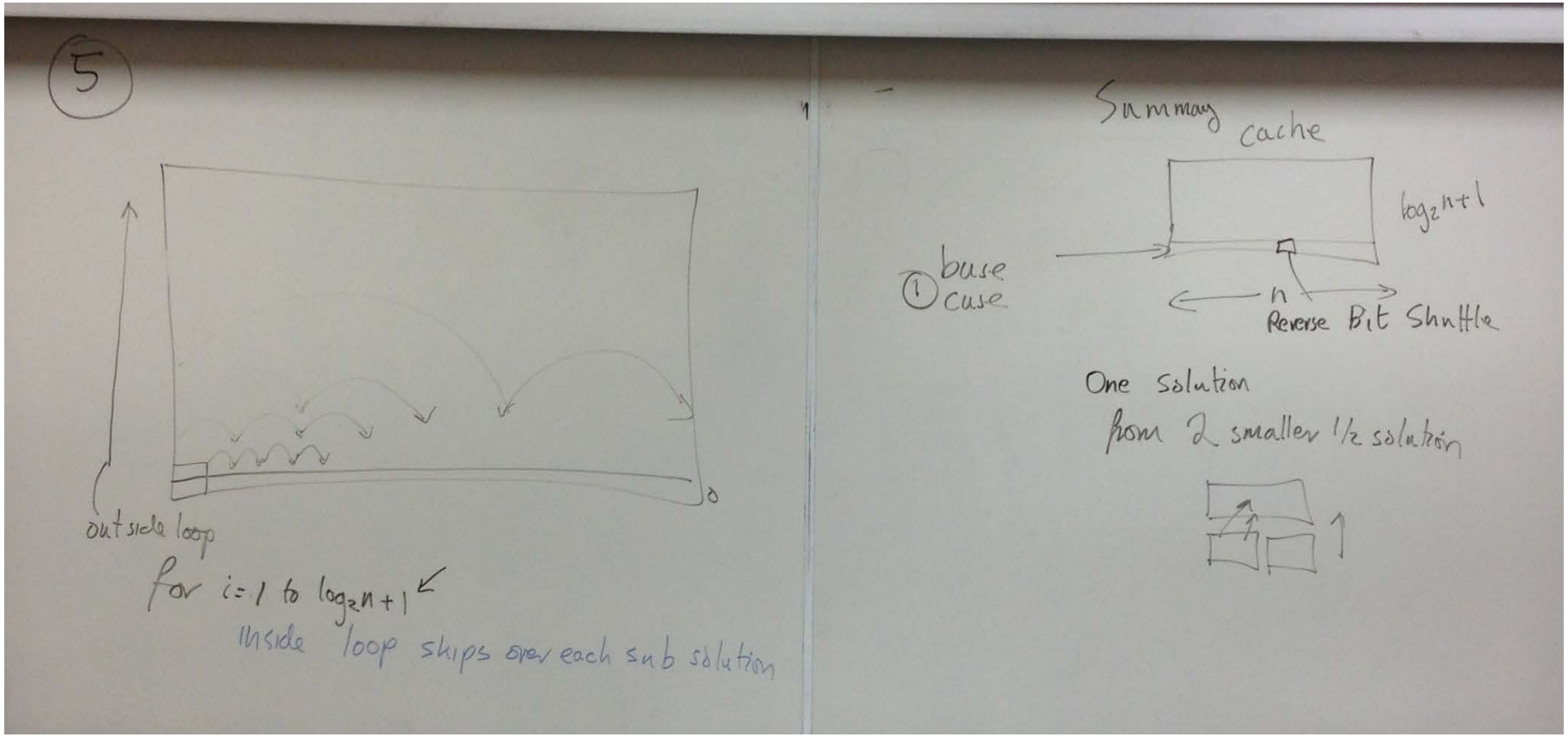

Base cases go in the FFTCache[i,0]. Reviewing the recursive def we see that each value is simply a coefficient of the polynomial. But the ith coefficient does not go into the ith place in the array!
What is the mapping from the index of the coefficient to the index of where it should go? ReverseBitFlipShuffle




Reverse bit flip shuffle example. For 8 coeficients, top line lists the coeficients
Next lines do the even (on the left) odd (on the right) split for each recursive call
The bottom line of numbers is where the coeficients have to be placed in the base case
DP cache
To determine the pattern, write the numbers in binary.
To compute the position of the ith coefficient: represent the number i in binary, reverse the bits, the value of this number is the location.




Study the solution construction process for one recursive call. We compute an array of size values from two sub solutions each of size/2 values. The subsolution on the left is for the evens, the one on the right is from the odds.




To complete the DP calculation, nested loops must be written that scan from bottom to top taking into account that the: (a) size of the problems increase from 2 to n, (b) the number of problems reduces from n/2 to 1.