

We have modified our code a decent bit from the original idea. The first and most significant change is the introduction of Graphics. We now have the functionality to print out our crossword as a graphical object. We also have a flag for toggling graphics. Next, we have also added flags for a set of preloaded wordlists. These include English language words, common names, acronyms, and places. Our algorithm for the creation of the crossword is where we ran into the most difficulty. We initially used a brute force method to place a word in our grid and then look for possible other words using regex to fill out the grid while still using words. This works for a 3x3 grid and a 4x4 grid. We have also tried to implement a Depth-first-Search algorithmic method, but that doesn't work consistently, although theoretically could be faster. The Depth-First-Search works best for 3x3 grids but it can do 4x4 grids for the "names" and "acronyms" wordbank. Otherwise, the brute force method should be used for any crossword that takes too long. There is a fundamental problem in our project in the user has no way of knowing when to stop the generation; we recommend to stop and restart any generation that takes more than a minute. We have included a flag in our input to indicate which method to use in the final product (DFS or brute force).