1. Specify the problem definition:

From the request of the guideline pdf file, I understand that : In this homework task, we have to access to some corpus of language and from that, creating a list of pairs of adjectives and verbs that are synonyms of each other but contrasted in intensity. The point here is that these words must be similar in their meanings, however, one word in the pair would have high intensity of expression while the other one has low intensity. Furthermore, one more task is to find out an intensity-modifying adverb to show the contrast of them in intensity, for example “highly” is the adverb for the pair “extol” and “praise”.

**C**.**a**. I would show my idea to prove why my output list is reasonable:

Firstly, I create a list of intensity-modifying adverbs(and I name this list “adverbs\_list” in the code) containing 15 adverbs in the list and each of them are very strong adverb, that mean adding these adverbs to a verb or an adjective can increase the expression intensely ( for example the adverb 'incredibly' ). After that, I access the ‘brown’ corpus and create a list of index of adjectives or verbs that is adjacent to one of adverbs in the adverbs\_list( for example, the code would pick the word “poor” from the text “This city is really poor” because it follows the word “really” that is contained in the adverbs\_list. After collecting these words, I create a list containing adjectives satisfying the condition and name it ‘list\_index\_after\_pos\_adj”, similarly a list named “list\_index\_after\_pos\_verb” for verbs. We need two separate lists for adjectives and verbs because adjectives are usually put after an adverb in text, while verbs are put before adverb.

In the text of “brown” corpus, if we have “…incredibly A”, then it means that the intensity of A is immediate or weak because it can combine with a strong adverb like “incredibly” to increase the intensity. The idea here is : words collected in 2 lists “list\_index\_after\_pos\_verb”, “list\_index\_after\_pos\_adj” have weak or immediate level of intensity, and next, from the “brown” corpus, we would find synonyms of each word from these 2 lists and furthermore, we would only collect synonyms with strong intensity. By this idea, we can create pairs containing one word with weak or immediate intensity while the other has strong level of intensity and 2 words in each pair are synonyms of each other.

Now the problem is how to measure the level of intensity of words to choose strongest ones. My idea for this problem is: I notice one point that words with strong intensity rarely combine with an adverb in the context, for example, because “destitute” means “very poor”, so it would be unreasonable if we have “ very destitute”. So, I create a function to return the score for how frequent a word combines with an adverb, and use this function to measure the intensity of word. The idea of this function is : assuming the index of the word A in context is pos\_A, then we would count the number of occurrence of A in the words list, and giving this value to the variable occurrence\_count. Next, let variable adverb\_attach\_count=0, and in case A is an adjective: for each occurrence of A, if the word at index (pos\_A-1) is adverb, then adverb\_attach\_count+=1 (because adjectives usually follow adverb), but in case A is a verb, if word at index (posA+1) is adverb, then adverb\_attach\_count+=1. Finally, this function returns the score (adverb\_attach\_cout/occurrence\_count). A word that more usually combine with adverb would have high score, while word with low score means it rarely combines with adverb. In program, we call this function adverb\_attach\_frequency(word). This function is used to measure intensity of word by score returned, higher score means weaker intensity.

Now, assuming, we have word A from one of 2 list “list\_index\_after\_pos\_verb”, “list\_index\_after\_pos\_adj”, then from “brown” corpus, we create a list containing all synonyms of A, for each synonym in this list, we get its score through function adverb\_attach\_frequency. Then, after getting score for all synonyms of A, the synonyms with lowest scores returned by adverb\_attach\_frequency would be the synonyms with strongest intensity of word A. (\*\*\*)

**b**. The idea to improve the algorithm in the future:

In the above explain of (\*\*\*), I use a variables threshold\_adj=0.1, threshold\_verb=0.1 to choose strong-intensity synonyms, that mean if the score returned by function adverb\_attach\_frequency for a synonym is lower than threshold, then it would be marked as strong and would be chosen. But I choose the threshold = 0.1 because I think it is small enough and using this result giving me good pairs at the end. But in fact, I believe that by more experiments, we can choose a better value for the threshold variable, to get best pairs. Secondly, the list adverb\_list is created manually by myself, but I believe that in English language, there are more intensity-modifying adverbs like that, and we can update them to the adverb\_list. Finally, I believe that there are more factors that can be used to measuring level of intensity of verb, adjective (for example the function adverb\_attach\_frequency bases on the frequency that a word combine with adverb to give score). If we can figure out more, I believe that the code would choose give a better list of strong-intensity synonym for each word.