Selen Erdoğan 210104004131 HM7

Sir, there are some points that I need to mention in the homework. Although my functions work separately, when I need to run them all together, I am faced with segmentation fault because my computer does not have enough memory. I couldn't use that method either because we haven't processed dynamic memory allocation yet and Yakup Hoca said that you can't use the subject in homework. The computer output is as follows. But my codes and the algorithm in my codes work in accordance with the pdf structure. Thank you

```
ulimit -s unlimited && ./test
Text contains 25796 words:
Dictionary contains 1000 words:
Similarity between car and bus: -1.000000
```

I had to make some changes to the makefile structure with the make run command in order to find the structure suitable for memory. And I had to use ulimit -s unlimited beforehand.

```
int histogram(const char words[][MAX_WORD_SIZE], const int occurrences[], char hist[][MAX_WORD_SIZE + 5 + 20]) {
    int max_occurrences = 0;
    int index = 0;
    while (strcmp(words[index], "-") != 0) {
        if (occurrences[index] > max_occurrences) {
            max_occurrences = occurrences[index];
        }
        index++;
    }

    int scale = max_occurrences > 20 ? max_occurrences / 20 : 1;
int j;
    index = 0;
    while (strcmp(words[index], "-") != 0) {
        int num_chars = occurrences[index] / scale;
        sprintf(hist[index], "%s: ", words[index]);
        for ( j = 0; j < num_chars; j++) {
            strcat(hist[index], "*");
        }
        index++;
    }

    strcpy(hist[index], "-");
    return scale;
}

int find_word_index(char *word, char dict[][MAX_WORD_SIZE], int dict_size) {
          int i;
    for ( i = 0; i < dict_size; i++) {
        if (strcmp(word, dict[i]) == 0) {
            return i;
        }
    }
    return -1;
    }
int find_word(const char words[][MAX_WORD_SIZE], const char *word) {
    int i = 0;
    while (strcmp(words[i], "-") != 0) {
        if (strcmp(words[i], word) == 0) {
            return i;
        }
        i++;
    }
    return -1;
```

```
}
int find_closest_word(const char words[][MAX_WORD_SIZE], const char *word, float threshold) {
    int closest_index = -1;
    double min_dissimilarity = 1000000.0; // buyuk bir baslangıc degeri
    int i = 0;

    int dict_size = 0;
    while (strcmp(words[dict_size], "-") != 0) {
        dict_size++;
    }

    float vectors[MAX_NUM_WORDS][MAX_VECTOR_SIZE]; // Veektorleri temsil eden float dizisi

    memset(vectors, 0, sizeof(float) * MAX_NUM_WORDS * MAX_VECTOR_SIZE);

    while (strcmp(words[i], "-") != 0) {
        double dissim = dissimilarity((char )words[i], (char *)word, vectors, (char ()[MAX_WORD_SIZE])words, dict_size);
        if (dissim < min_dissimilarity) {
            min_dissimilarity = dissim;
            closest_index = i;
        }
        i++;
    }

    if (min_dissimilarity <= threshold) {
        return closest_index;
    } else {
        return -1;
    }
    }
*/
```

I set up the necessary algorithm structure for the histogram function to perform its function well enough. Function structures that need to be written extra are also ready. But in order for the code to output, I had to put it in the comment line, Sir. I apologize. My other functions are work together.