מחשוב מקבילי ומבוזר

תרגיל #3

The purpose of this exercise is to implement a simple parallel application with OpenMP or CUDA and compare the execution times

Parallelize the following code:

```
#include <stdio.h>
#include <math.h>
#include <stdlib.h>


// This function performs heavy computations
// No Changes to this function are allowed
 double heavy(double data, int loopSize) {
    double sum = 0;
    for (int i = 0; i < loopSize; i++)
        sum += cos(exp(sin(data * (i % 11))))/ loopSize;

     return sum;
}


// Sequential code to be parallelized
int main(int argc, char *argv[]) {
    int i;
    int size = atoi(argv[1]);
    int loopSize = atoi(argv[2]);
    double* arr = (double*)malloc(size * sizeof(double));

    for (i = 0; i < size; i++)
        arr[i] = rand() / RAND_MAX;

    double answer = 0;
    for (i = 0; i < size; i++)
        answer += heavy(arr[i], loopSize);

    printf("answer = %e\n", answer);
}
```

Requirements:

1. Implement two approaches to parallelize the code:
    a. The first solution uses **OpenMP** for parallelization
    b. The second solution uses **CUDA** for parallelization

2. Run, measure execution time, explain the results. The table with the time measurement and your explanation of the results are to be placed in the separate Word file named **results.doc** in the root directory of the solution.

3. No changes to function heavy are allowed.

Parameters		**Execution time**		
size	**loopSize**	**Sequential**	**OpenMP**	**CUDA**
1000	1000	0.028s	0.025s	
1000	10000	0.298s	0.257s	
10000	1000	0.307s	0.228s	
10000	10000	2.781s	2.103s	
100000	100000	281.257s	197.577s	

Grading Policy:

- **10 points** for code quality:
    a. The code must be divided into small functions (not more than 40 lines of code).
    b. Use meaningful names for variables, functions, files, constants.
    c. Place enough comments to understand the code
    d. No unused lines of code. Don't repeat the code – use functions!
    e. Write README.TXT file if special instructions are needed to run the solution. The file must be in the root folder of the solution.
- **70 points** – for proper implementation of parts 1, 2 of the requirements.

- **20 points** – for final results explanation and for time measurement.

- The Homework must be delivered in time. No delay will be accepted.

Important:

- The homework may be performed in pairs. Only one member of pair submits the solution through the Moodle. The whole project must be zipped and named as

111111111_222222222.zip

Where **111111111** is ID of the one student and **222222222** is ID of another student

בהצלחה

בהצלחה!