Gebze Institute of Technology
Introduction to Programming
CSE102 HW06
Fall 2014

Due Date
06.11.2014, 23:59

Implement the following functions;

1. **(10 pts)** `void fill_vector(double arr[], int* size);`
   Reads content of the given input array from the user via console.
   input format:
   (<size> < …. Data…. >)
   3 1.5 34.5 4.4
2. **(10 pts)** `void print_vector(double arr[], int size);`
   Prints content of the given input array. (output format: [1.5, 34.5, 4.4])
3. **(10 pts)**
   // *result = arr1 + arr2*
   ```
   void add_vector(double arr1[], double arr2[],int size_arr1,
   size_arr2);
   ```

   Adds elements of arr1 and arr2 and prints the resulting array to the console.
   (output format:
   Add: [1.5, 34.5, 4.4]
)
4. **(10 pts)**
   // *result = arr1 - arr2*
   ```
   void sub_vector(double arr1[], double arr2[],int size_arr1,
   size_arr2);
   ```
   Subtracts elements of arr2 from arr1 and prints the resulting array to the console.
   (output format:
   Sub: [1.5, 34.5, 4.4]
)
5. **(10 pts)**
   // *result = mag(arr1)*
   `void mag (double arr1[],int size_arr1);`
   Computes magnitude of arr1 and prints the resulting values to the console.
   (output format:
   Mag: 1.5
)
6. **(20 pts)**
   // *result[i] = dist(arr1[i], arr2[i])*
   ```
   void dist(double arr1_x[],double arr1_y[],double
   arr1_z[],double arr2_x[],double arr2_y[],double arr2_z[], int
   size_arr1, size_arr2);
   ```
   Computes 3D Euclidian distance between each 3D point of arr1 and arr2. Then, prints the

resulting 3D points to the console.
 (output format:
Dist: [1.5, 34.5, 4.4]
)

7. **(30 pts)**
   // *result = most_distant (arr1)*
   `void most_distant(double arr1_x[],double arr1_y[],double arr1_z[], int size_arr1);`
   Finds two most distant 3D points in arr1 and prints the maximum distance to the console.
    (output format:
   Max Dist: 400.4
)

Maximum input size is 1000.

Your main function must be in following format;

```
int main(){
    //You have to complete the code and correct all kind of errors

    ///////////////////////////////////////
    puts(“-------------------------------”);
    printf(“testing the function xxx \n”);

    ...call fill_vector and print_vector...
    //sample input:
    //3 1.5 34.5 4.4

    puts(“-------------------------------”);
    ///////////////////////////////////////


    ///////////////////////////////////////
    puts(“-------------------------------”);
    printf(“testing the function xxx \n”);

    ...call fill_vector and add_vector...
    //sample input: arr1, arr2
    //3 1.5 34.5 4.4
    //3 4.5 54.5 7.4

    puts(“-------------------------------”);
    ///////////////////////////////////////


    ///////////////////////////////////////
    puts(“-------------------------------”);
    printf(“testing the function xxx \n”);
```

```
...call fill_vector and sub_vector...
//sample input:
//3 1.5 34.5 4.4
//3 4.5 54.5 7.4

puts("-------------------------------");
////////////////////////////////////////




////////////////////////////////////////
puts("-------------------------------");
printf("testing the function xxx \n");

...call fill_vector and mag...
//sample input:
//3 1.5 34.5 4.4

puts("-------------------------------");
////////////////////////////////////////




////////////////////////////////////////
puts("-------------------------------");
printf("testing the function xxx \n");

...call fill_vector and dist...
//sample input: arr1_x, arr1_y, arr1_z, arr2_x, arr2_y, arr2_z
//3 1.5 34.5 4.4
//3 4.5 54.5 7.4
//3 6.5 64.5 8.4

//3 19.5 34.5 4.4
//3 49.5 54.5 7.4
//3 69.5 64.5 8.4


puts("-------------------------------");
////////////////////////////////////////



////////////////////////////////////////
puts("-------------------------------");
printf("testing the function xxx \n");

...call fill_vector and most_distant...
//sample input: arr1_x, arr1_y, arr1_z
//3 1.5 34.5 4.4
//3 4.5 54.5 7.4
//3 6.5 64.5 8.4
```

```
    puts(“-------------------------------”);
    ////////////////////////////////////////

}
```

Notes:

- You should submit 1 file;
  - main.c
- Add all **<u>files into a folder and compress it</u>** for submission. The folder names will be restricted to the following format:
  HW#_studentid_studentname.
  - Example:
    HW01_121044001_Abdullah_Akay
- Upload soft copy of your homework to Moodle course web page
- **<u>DON’T submit hard copy of your assignment</u>**.
- Don’t forget to test your code in the provided Linux virtual machine.
- Obey good programming rules (Indentation, Documenting, Well Commenting, Avoidıng magic numbers, Non-ascii characters etc.)
- **<u>Strictly follow submission and file, folder naming rules.</u>**