İTÜGAE
GÜNEŞ ARABASI EKİBİ
2004

powered by
ZES
Zorlu Energy Solutions

ISTANBUL TECHNICAL UNIVERSITY

ITU ZES Solar Car Team

2024/2025

Weekly Assignment-1

07.12.2023

1 Point Structure

You are expected to implement Point structure using C++ structs. It must have 3 float attributes, x, y, and z. And you need to implement various functions related to this struct:

- zero_distance() -> distance to (0, 0, 0)
- distance() -> takes two Point and return the distance
- compare() -> takes two Points and returns which one is further from zero
- region() -> return region of the Point in 3D space
- is_in_same_region() -> returns whether two Points are in the same region or not

Design is up to you, function names above are just for example and may differ depending on your implementation. You are expected to write just main.cpp. For better implementation maybe you implement a Region enum object.

2 Calculator Class Implementation

You are expected to implement a basic Calculator class with user interaction via terminal. This class contains just the necessary methods:

- addition()
- subtraction()
- multiplication()
- division() -> be careful to zero division
- square()
- exponentiation()
- modulus()

You must submit 3 different files, which are Calculator.h, Calculator.cpp, main.cpp. Implement error handling for invalid user inputs (e.g., non-numeric input). Provide an option for the user to perform multiple calculations in one session without exiting the program. You need to read the user's input and print the results to the user. For better implementation maybe you can use throw-catch structure in C++ for error handling. For now, you can implement just using floats but it is better to implement class as a template class.

3 Matrix Class Implementation

You are expected to implement a functional Matrix class with its methods. You must create Matrix class and you can use std::vector for implementing operations. You need to hold your matrix as a private attribute. Your class has to have these methods:

- constructor -> You need to take the dimensions and entity of the whole matrix, by default you need to create a zero matrix or you can give std::vector as an argument(constructor overloading)

- inverse() for 3x3 matrices at most
- trace()
- determinant() for 3x3 matrices at most
- transpose()
- add()
- subtract()
- neg()
- multiply() -> for pairwise multiplication
- dot() -> for dot multiplication
- magnitude()
- zeroes, ones, identity matrix creators
- utility functions and getters and setters if needed

Design is up to you, function names above are just for example and may differ depending on your implementation. You are expected to write 3 different files, which are matrix.h, matrix.cpp, main.cpp. Your tests must be in main.cpp and you are expected to use all the functions you implemented. For better programming exercises you need to create your Matrix class with integer vectors. Also, you must put your whole code into a namespace, the name is up to you, and reach the attributes over this namespace in main.cpp. You must use the const keyword as much as possible and do not forget to use pass by reference.

Further improvements:

- You can implement addition and subtraction methods by operator overloading
- You can implement Matrix class as a template class so that it works type diagnostic
- You can implement std::cout overload to print the Matrix
- You can use CMakeList.txt to build your package
- You can share your work on Github and write a documentation

These further improvements are for future insight into your assignment. You will be asked to implement these improvements later and update your work on GitHub.

4 Conclusion

For any questions and errors you find, contact with Nisanur Çontay. While preparing your work consider readability and scalability, and also do not forget to add necessary comments to your codes.