**[Code\_for\_the\_energy\_efficiency\_in\_open\_MPI\_for\_using\_parallel\_computing\_for\_HPC Problems](https://github.com/KatamVital/Code_for_the_energy_efficiency_in_open_MPI_for_using_parallel_computing_for_HPC_Problems/upload/main)**

**#include <mpi.h>**

**#include <stdio.h>**

**int main(int argc, char\*\* argv) {**

**MPI\_Init(&argc, &argv);**

**double start\_time, end\_time;**

**start\_time = MPI\_Wtime();**

**end\_time = MPI\_Wtime();**

**MPI\_Finalize();**

**double runtime = end\_time - start\_time;**

**printf("Runtime: %f seconds\n", runtime);**

**return 0;**

**}**

**Example for this :**

**Code for finding the differentation of x^3+5x^2+6x+2 at x=2 , using parallel computing to reduce the time taken to solve it :**

**#include <mpi.h>**

**#include <stdio.h>**

**double compute\_derivative(double x) {**

**return 3 \* x \* x + 10 \* x + 6; // Derivative of x^3 + 5x^2 + 6x + 2**

**}**

**int main(int argc, char\*\* argv) {**

**MPI\_Init(&argc, &argv);**

**double start\_time, end\_time;**

**start\_time = MPI\_Wtime();**

**double x = 2.0;**

**double derivative = compute\_derivative(x);**

**end\_time = MPI\_Wtime();**

**MPI\_Finalize();**

**double runtime = end\_time - start\_time;**

**printf("Derivative at x = %f: %f\n", x, derivative);**

**printf("Runtime: %f seconds\n", runtime);**

**return 0;**

**}**

**Output :**