FEDERAL STATE AUTONOMOUS EDUCATIONAL INSTITUTION OF HIGHER EDUCATION

ITMO UNIVERSITY

Report

on the practical task No. 3,4,5

Performed by

*Alexander Yamoldin*

*J4134c*

St. Petersburg 2021

# Goal

Understand basic MPI C++ syntax, use MPI library to print messages from difference process, to understanding the order of printing message and to determination of execution time.

# Formulation of the problem

In the Assignment 3 need to compile and run Assignment3.c program.

In the Assignment 4 need to convert the code Assignment4.c to match your individual version of the assignment. Your task number is your number in the reporting table <https://docs.google.com/spreadsheets/d/1m2KqmSc5pMMQ9JezZGybnuKG75FywUcs8gbPuMvVUxc>

In my case the number of task is 21.

#21

The root process accepts messages from child processes and determines whether the sequence is strictly descending.

In the Assignment 5 need to compile and run Assignment5.c program.

Explain in detail how it works.

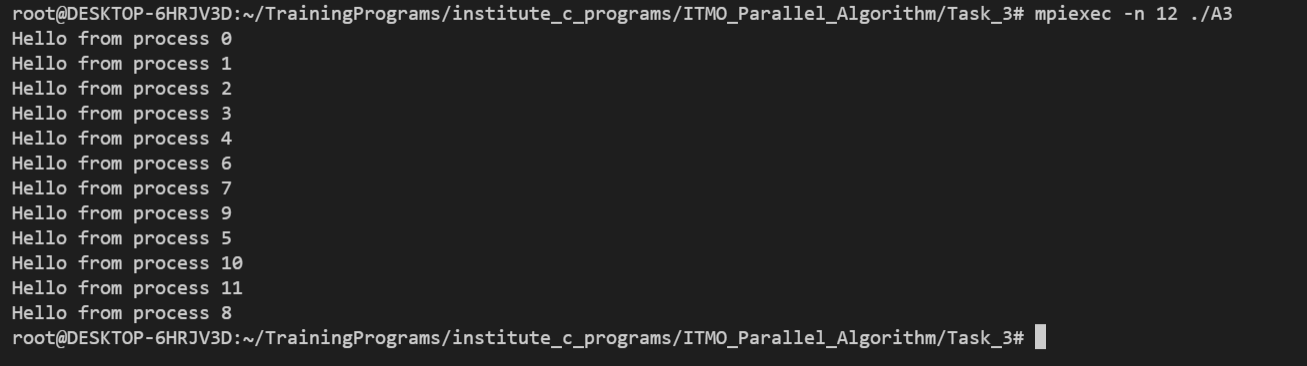
Determine the execution time of the program from the previous task.

# Results

The code of the Assignment 3 can be found in

<https://github.com/AAYamoldin/TrainingPrograms/blob/master/institute_c_programs/ITMO_Parallel_Algorithm/Task_3/Assignment3.c>

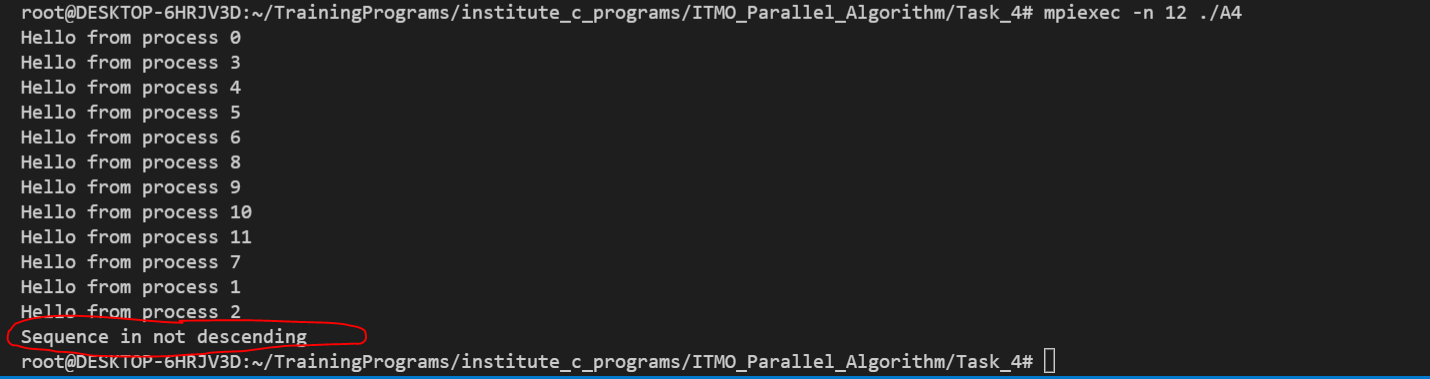
The result of the program in the picture below:



The code of the Assignment 4 can be found in

<https://github.com/AAYamoldin/TrainingPrograms/blob/master/institute_c_programs/ITMO_Parallel_Algorithm/Task_4/Assignment4.c>

The result of the program is the picture below:



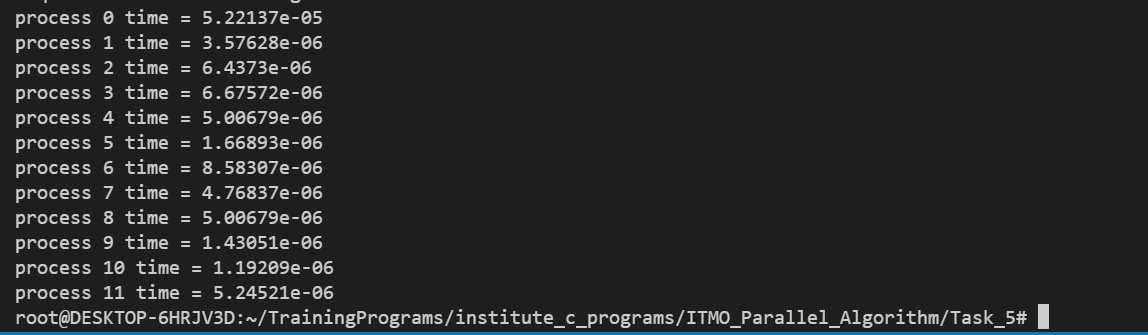
The code of the Assignment 5 can be found in

<https://github.com/AAYamoldin/TrainingPrograms/blob/master/institute_c_programs/ITMO_Parallel_Algorithm/Task_5/Assignment5.c>

and the code of second part of the Assignment5

<https://github.com/AAYamoldin/TrainingPrograms/blob/master/institute_c_programs/ITMO_Parallel_Algorithm/Task_5/time_complexity_assignment4.cpp>

The result of the program is in the picture below:



# Conclusions

In Assignment 3 we understanding how MPI works, we got acquainted with MPI paradigm such MPI\_Init for starting parallel form, MPI\_Comm\_size, MPI\_Comm\_rank the features of commutators, MPI\_Send, MPI\_Recv for messaging passing.

In Assignment 4 we define the type of the messages sequence and we define that our sequence is not descending. Also we can see that we have a race conditional (messages print in randomize order).

In the Assignment 5 we understanding how to measure execution time of the MPI process and measuring execution time of each process in Assignment 4.