Saint Petersburg National Research University of Information Technologies, Mechanics and Optics (ITMO University)

REPORT

about laboratory works

Assignment 16.
Assignment 17.
Assignment 18.
Assignment 19.
Assignment 20.
Assignment 21.

Student Pogrebnoy D.A. j4132c

Saint-Petersburg, 2021

ASSIGNMENT 16.

Task

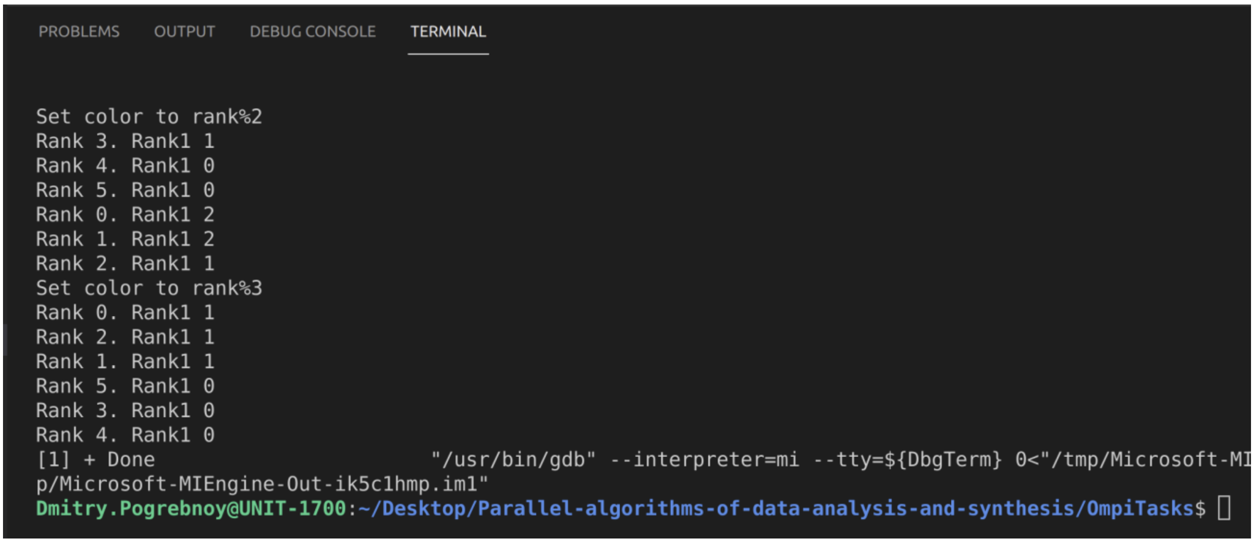
In the MPI_Comm_split function (Assignment16.c), replace the color parameter with (rank% 2), (rank% 3), look at how many groups the processes are split into, depending on the specified attribute of division into groups.

Implementation

Source code and data gathered are available on https://github.com/DmitryPogrebnoy/Parallel-algorithms-of-data-analysis-and-synthesis/blob/master/OmpiTasks/Task16/Assignment16.cpp

The description of the code is described in the comments.

Output example:



```
Set color to rank%2
Rank 3. Rank1 1
Rank 4. Rank1 0
Rank 5. Rank1 0
Rank 0. Rank1 2
Rank 1. Rank1 2
Rank 2. Rank1 1
Set color to rank%3
Rank 0. Rank1 1
Rank 2. Rank1 1
Rank 1. Rank1 1
Rank 5. Rank1 0
Rank 3. Rank1 0
Rank 4. Rank1 0
[1] + Done                       "/usr/bin/gdb" --interpreter=mi --tty=${DbgTerm} 0<"/tmp/Microsoft-MI
p/Microsoft-MIEngine-Out-ik5c1hmp.im1"
Dmitry.Pogrebnoy@UNIT-1700:~/Desktop/Parallel-algorithms-of-data-analysis-and-synthesis/OmpiTasks$ 
```

ASSIGMENTS 17.

Task

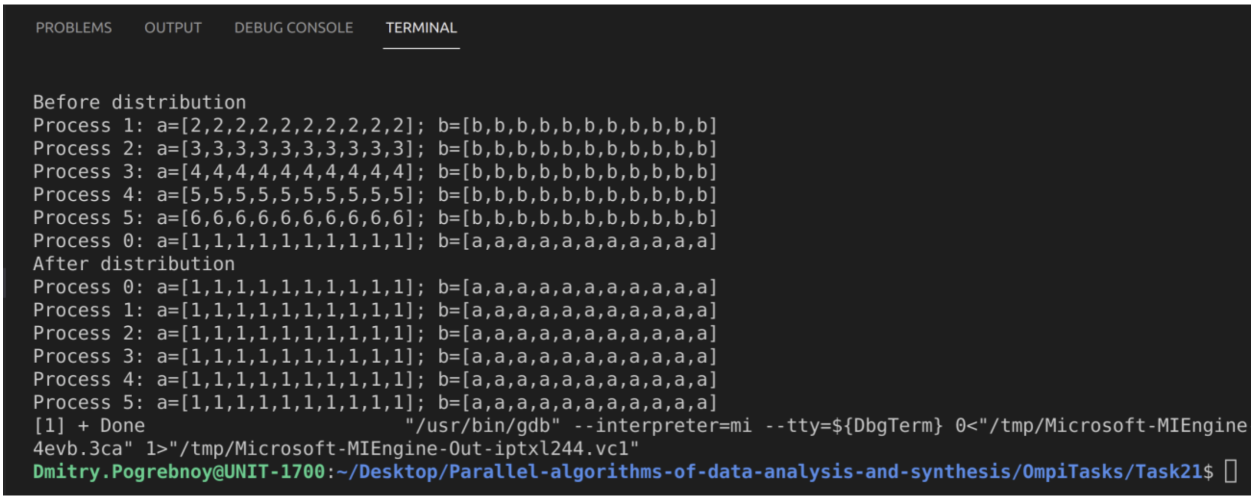
Understand the new functions in Assignment17.c. and explain program execution.
Display the values of the process number and arrays a[i], b[i], before packing and distribution, and after. See how broadcasting works.

Implementation

Source code and data gathered are available on https://github.com/DmitryPogrebnoy/Parallel-algorithms-of-data-analysis-and-synthesis/blob/master/OmpiTasks/Task17/Assignment17.cpp

The description of the code is described in the comments.

Output example:



```
Before distribution
Process 1: a=[2,2,2,2,2,2,2,2,2,2]; b=[b,b,b,b,b,b,b,b,b,b,b]
Process 2: a=[3,3,3,3,3,3,3,3,3,3]; b=[b,b,b,b,b,b,b,b,b,b,b]
Process 3: a=[4,4,4,4,4,4,4,4,4,4]; b=[b,b,b,b,b,b,b,b,b,b,b]
Process 4: a=[5,5,5,5,5,5,5,5,5,5]; b=[b,b,b,b,b,b,b,b,b,b,b]
Process 5: a=[6,6,6,6,6,6,6,6,6,6]; b=[b,b,b,b,b,b,b,b,b,b,b]
Process 0: a=[1,1,1,1,1,1,1,1,1,1]; b=[a,a,a,a,a,a,a,a,a,a,a]
After distribution
Process 0: a=[1,1,1,1,1,1,1,1,1,1]; b=[a,a,a,a,a,a,a,a,a,a,a]
Process 1: a=[1,1,1,1,1,1,1,1,1,1]; b=[a,a,a,a,a,a,a,a,a,a,a]
Process 2: a=[1,1,1,1,1,1,1,1,1,1]; b=[a,a,a,a,a,a,a,a,a,a,a]
Process 3: a=[1,1,1,1,1,1,1,1,1,1]; b=[a,a,a,a,a,a,a,a,a,a,a]
Process 4: a=[1,1,1,1,1,1,1,1,1,1]; b=[a,a,a,a,a,a,a,a,a,a,a]
Process 5: a=[1,1,1,1,1,1,1,1,1,1]; b=[a,a,a,a,a,a,a,a,a,a,a]
[1] + Done                    "/usr/bin/gdb" --interpreter=mi --tty=${DbgTerm} 0<"/tmp/Microsoft-MIEngine
4evb.3ca" 1>"/tmp/Microsoft-MIEngine-Out-iptxl244.vc1"
Dmitry.Pogrebnoy@UNIT-1700:~/Desktop/Parallel-algorithms-of-data-analysis-and-synthesis/OmpiTasks/Task21$ 
```

ASSIGNMENT 18.

Task

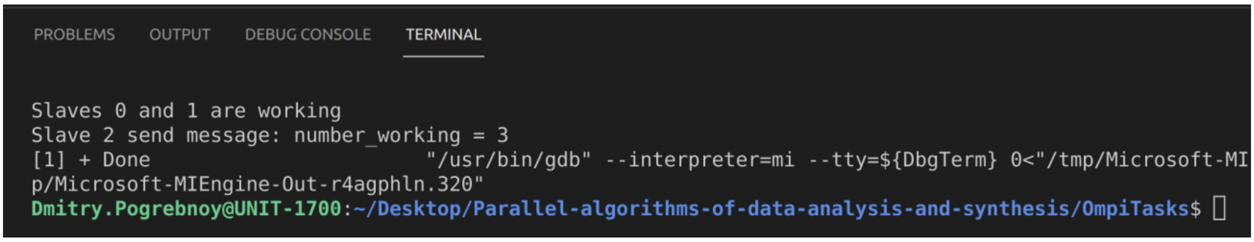
To complete the task, you need to create and compile two programs: Master (master.o) and Slave (slave.o). The Master should start the worker, so be careful with the names of the executable files.
Launch the master via the mpiexec command for one process.
Startup example: mpiexec -n 1 ./master.o
Understand the new functions in Assignment18_master.c and Assignment18_slave.c and explain programs execution.
Add a third process, which will transfer from the slave processes to the master the number of running processes, the master should receive and display.

Implementation

Source code and data gathered are available on https://github.com/DmitryPogrebnoy/Parallel-algorithms-of-data-analysis-and-synthesis/tree/master/OmpiTasks/Task18

The description of the code is described in the comments.

Output of implemented task example:



```
Slaves 0 and 1 are working
Slave 2 send message: number_working = 3
[1] + Done                       "/usr/bin/gdb" --interpreter=mi --tty=${DbgTerm} 0<"/tmp/Microsoft-MI
p/Microsoft-MIEngine-Out-r4agphln.320"
Dmitry.Pogrebnoy@UNIT-1700:~/Desktop/Parallel-algorithms-of-data-analysis-and-synthesis/OmpiTasks$ 
```

ASSIGNMENT 19.

Task

To complete the task, you need to create and compile two programs: server and client. In one window of the SSH client, a server is launched for one process, which gives out the port name.
An example of a command to start the server: mpiexec -n 1 ./serv.o
Then the client is launched in another window, specifying the port name separated by a space in single quotes (example command: mpiexec -n 1 ./client.o ‘port name’).
Understand the new functions in Assignment19_serv.c and Assignment19_client.c and explain programs execution.
Check the work by running the server and the client. Add the program and send an arbitrary message to each other.

Implementation

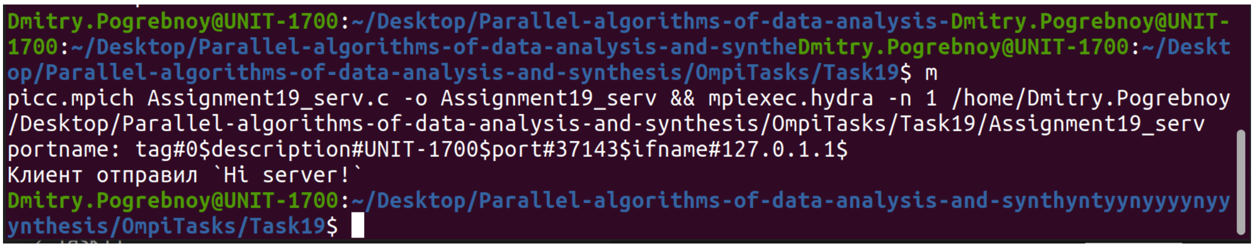
Source code and data gathered are available on

https://github.com/DmitryPogrebnoy/Parallel-algorithms-of-data-analysis-and-synthesis/tree/master/OmpiTasks/Task19

The description of the code is described in the comments.

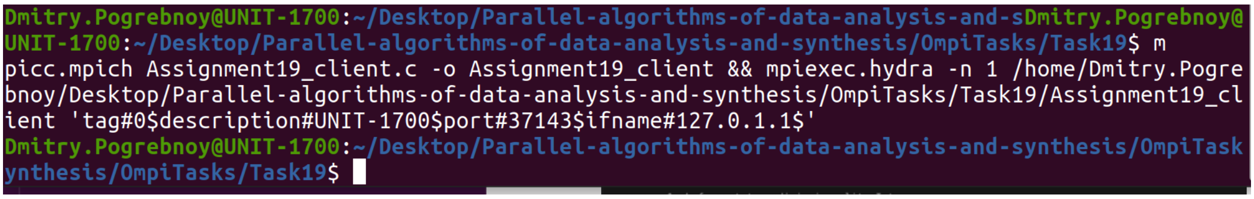
OpenMPI is broken, so I used MPICH to compile and run the source code.

Run server:



```
Dmitry.Pogrebnoy@UNIT-1700:~/Desktop/Parallel-algorithms-of-data-analysis-Dmitry.Pogrebnoy@UNIT-
1700:~/Desktop/Parallel-algorithms-of-data-analysis-and-syntheDmitry.Pogrebnoy@UNIT-1700:~/Deskt
op/Parallel-algorithms-of-data-analysis-and-synthesis/OmpiTasks/Task19$ m
picc.mpich Assignment19_serv.c -o Assignment19_serv && mpiexec.hydra -n 1 /home/Dmitry.Pogrebnoy
/Desktop/Parallel-algorithms-of-data-analysis-and-synthesis/OmpiTasks/Task19/Assignment19_serv
portname: tag#0$description#UNIT-1700$port#37143$ifname#127.0.1.1$
Клиент отправил `Hi server!`
Dmitry.Pogrebnoy@UNIT-1700:~/Desktop/Parallel-algorithms-of-data-analysis-and-synthyntyynyyyynyy
ynthesis/OmpiTasks/Task19$ 
```

Run client:



```
Dmitry.Pogrebnoy@UNIT-1700:~/Desktop/Parallel-algorithms-of-data-analysis-and-sDmitry.Pogrebnoy@
UNIT-1700:~/Desktop/Parallel-algorithms-of-data-analysis-and-synthesis/OmpiTasks/Task19$ m
picc.mpich Assignment19_client.c -o Assignment19_client && mpiexec.hydra -n 1 /home/Dmitry.Pogre
bnoy/Desktop/Parallel-algorithms-of-data-analysis-and-synthesis/OmpiTasks/Task19/Assignment19_cl
ient 'tag#0$description#UNIT-1700$port#37143$ifname#127.0.1.1$'
Dmitry.Pogrebnoy@UNIT-1700:~/Desktop/Parallel-algorithms-of-data-analysis-and-synthesis/OmpiTask
ynthesis/OmpiTasks/Task19$
```

ASSIGNMENT 20.

Task

Understand the new functions in Assignment20.c, complete the program according to the assignment, explain the execution of the program.
Write a function that will create a file "file.txt" with random content (or with specific text). The function must be executed before the program reads the contents of the file. Run the program on one process. Check if the contents of the file are displayed correctly. Add an option that will delete the file on close.

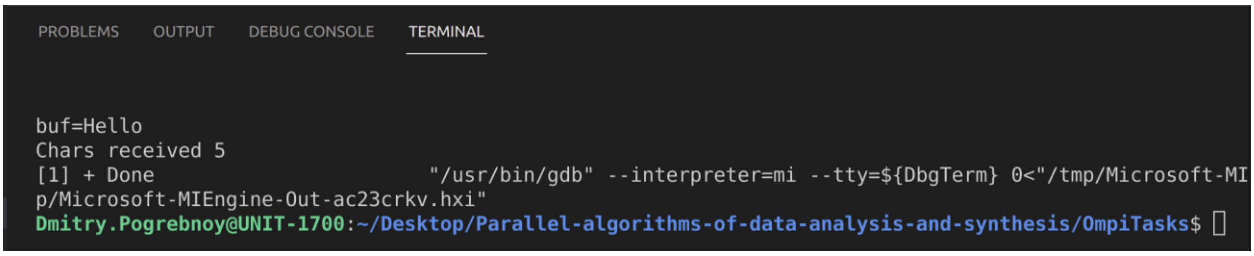
Implementation

Source code and data gathered are available on

https://github.com/DmitryPogrebnoy/Parallel-algorithms-of-data-analysis-and-synthesis/blob/master/OmpiTasks/Task20/Assignment20.c

The description of the code is described in the comments.

Output of implemented task example:



```
buf=Hello
Chars received 5
[1] + Done                       "/usr/bin/gdb" --interpreter=mi --tty=${DbgTerm} 0<"/tmp/Microsoft-MI
p/Microsoft-MIEngine-Out-ac23crkv.hxi"
Dmitry.Pogrebnoy@UNIT-1700:~/Desktop/Parallel-algorithms-of-data-analysis-and-synthesis/OmpiTasks$
```

ASSIGNMENT 21.

Task

Understand the new functions in Assignment21.c, complete the program according to the assignment, explain the execution of the program.

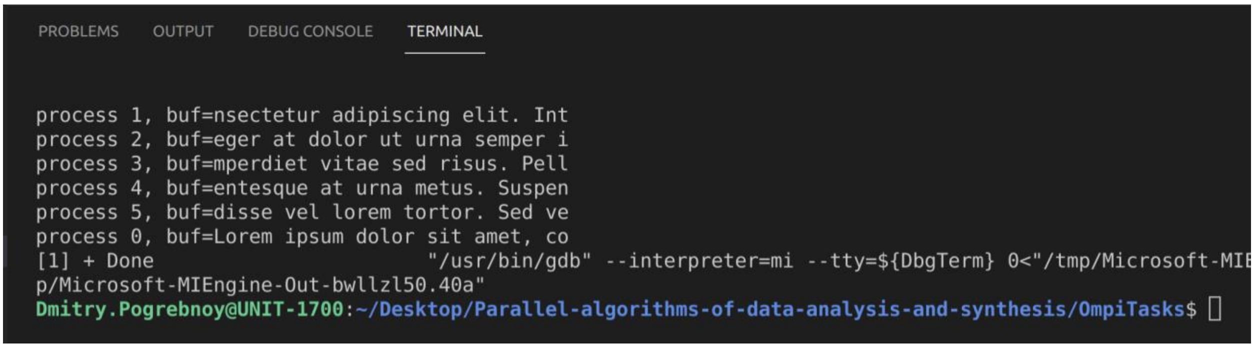
Create a file and fill it with bulky text, output the content in parallel. Change the step of reading the contents of the file and the number of characters to be output by each process.

Implementation

Source code and data gathered are available on https://github.com/DmitryPogrebnoy/Parallel-algorithms-of-data-analysis-and-synthesis/tree/master/OmpiTasks/Task21

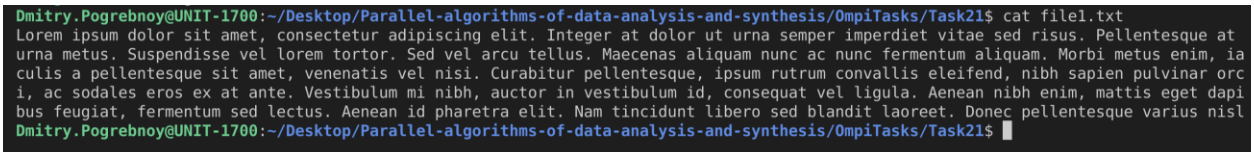
The description of the code is described in the comments.

Output of implemented task example:



```
process 1, buf=nsectetur adipiscing elit. Int
process 2, buf=eger at dolor ut urna semper i
process 3, buf=mperdiet vitae sed risus. Pell
process 4, buf=entesque at urna metus. Suspen
process 5, buf=disse vel lorem tortor. Sed ve
process 0, buf=Lorem ipsum dolor sit amet, co
[1] + Done                       "/usr/bin/gdb" --interpreter=mi --tty=${DbgTerm} 0<"/tmp/Microsoft-MIE
p/Microsoft-MIEngine-Out-bwllzl50.40a"
Dmitry.Pogrebnoy@UNIT-1700:~/Desktop/Parallel-algorithms-of-data-analysis-and-synthesis/OmpiTasks$ 
```

And file contains following data:



```
Dmitry.Pogrebnoy@UNIT-1700:~/Desktop/Parallel-algorithms-of-data-analysis-and-synthesis/OmpiTasks/Task21$ cat file1.txt
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Integer at dolor ut urna semper imperdiet vitae sed risus. Pellentesque at
urna metus. Suspendisse vel lorem tortor. Sed vel arcu tellus. Maecenas aliquam nunc ac nunc fermentum aliquam. Morbi metus enim, ia
culis a pellentesque sit amet, venenatis vel nisi. Curabitur pellentesque, ipsum rutrum convallis eleifend, nibh sapien pulvinar orc
i, ac sodales eros ex at ante. Vestibulum mi nibh, auctor in vestibulum id, consequat vel ligula. Aenean nibh enim, mattis eget dapi
bus feugiat, fermentum sed lectus. Aenean id pharetra elit. Nam tincidunt libero sed blandit laoreet. Donec pellentesque varius nisl
Dmitry.Pogrebnoy@UNIT-1700:~/Desktop/Parallel-algorithms-of-data-analysis-and-synthesis/OmpiTasks/Task21$ 
```