CSE 4001

PARALLEL AND DISTRIBUTED COMPUTING

Practice Session

L27+L28 | PLBG04
December 3, 2021
Dr. Narayanan Prasanth

FALL SEMESTER 2021-22

by

SHARADINDU ADHIKARI
19BCE2105

Question 1:

Problem - 1

Given the Date of Birth of Mr. Raju, Calculate the following with respect to today's date.

1. Number of birthdays
2. Number of days
3. Number of weeks

While calculating the birthday, If Mr. Raja was born on 29th February of a leap year then he celebrates his birthday only in leap years.

Implement the above requirements serially and parallelly (using OpenMP components) and print the time taken in both the cases. Write down the aim and algorithm for the problem.

Test Cases (min 3 req.)
Input 1: 29.2.1940
Input 2: 20.2.2000
Input 3: "your choice"

Code:

Apologies Sir. Couldn't answer this question.

Question 2:

Problem - 2

Develop an MPI program to split a single global communicator into a set of smaller communicators based on the color. Print the old rank, new rank, old size and new size of the processes.

Code:

P2-Split.c

```c
#include <stdlib.h>
#include <stdio.h>
#include <mpi.h>
```

sharadindu.adhikari2019@vitstudent.ac.in

```c
int main(int argc, char **argv) {

  printf("\nSharadindu Adhikari, 19BCE2105 \n");
  printf("\n \n"); //for some space

  MPI_Init(NULL, NULL);

  // Get the rank and size in the original communicator
  int world_rank, world_size;
  MPI_Comm_rank(MPI_COMM_WORLD, &world_rank);
  MPI_Comm_size(MPI_COMM_WORLD, &world_size);

  int color = world_rank / 4; // Determine color based on row

  // Split the communicator based on the color and use the original rank for
ordering
  MPI_Comm row_comm;
  MPI_Comm_split(MPI_COMM_WORLD, color, world_rank, &row_comm);

  int row_rank, row_size;
  MPI_Comm_rank(row_comm, &row_rank);
  MPI_Comm_size(row_comm, &row_size);

  printf("WORLD RANK/SIZE: %d/%d --- ROW RANK/SIZE: %d/%d\n",
    world_rank, world_size, row_rank, row_size);

  MPI_Comm_free(&row_comm);

  MPI_Finalize();
}
```
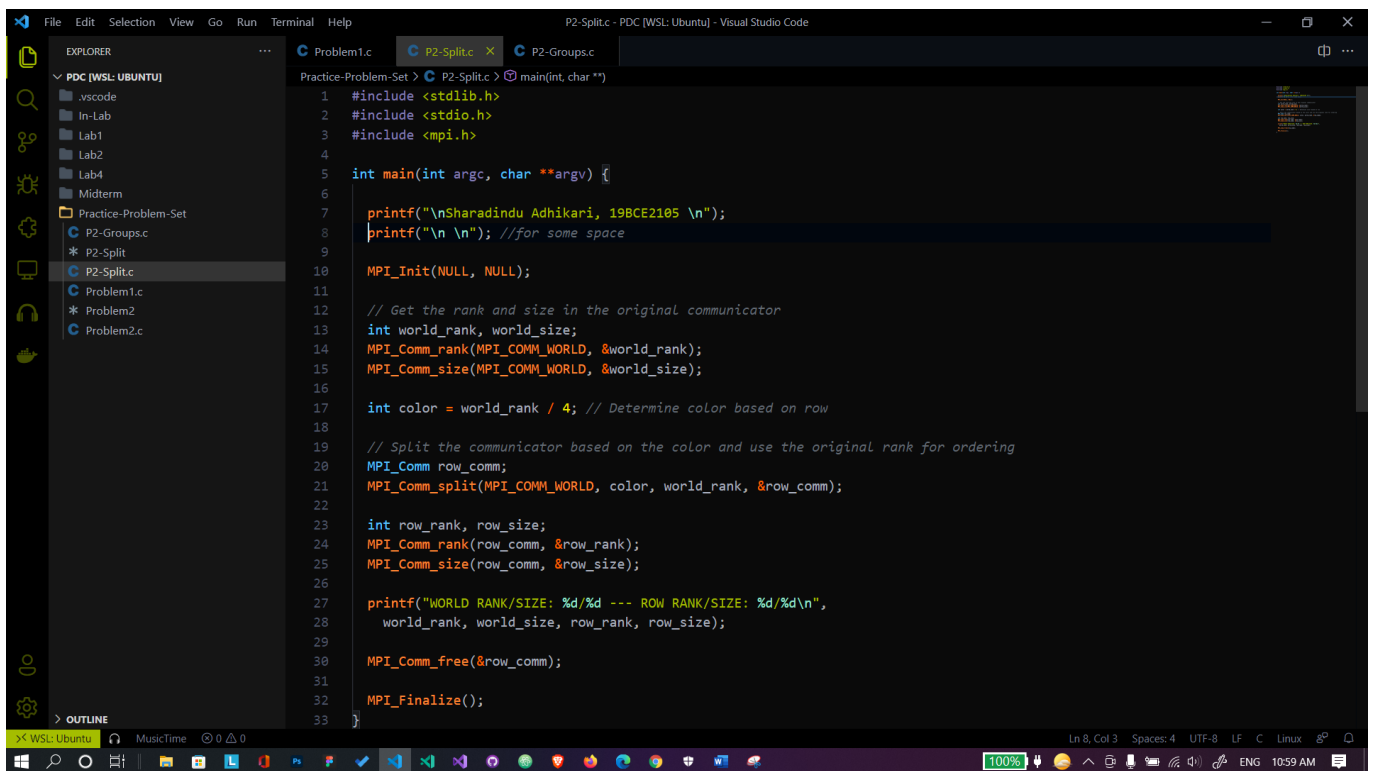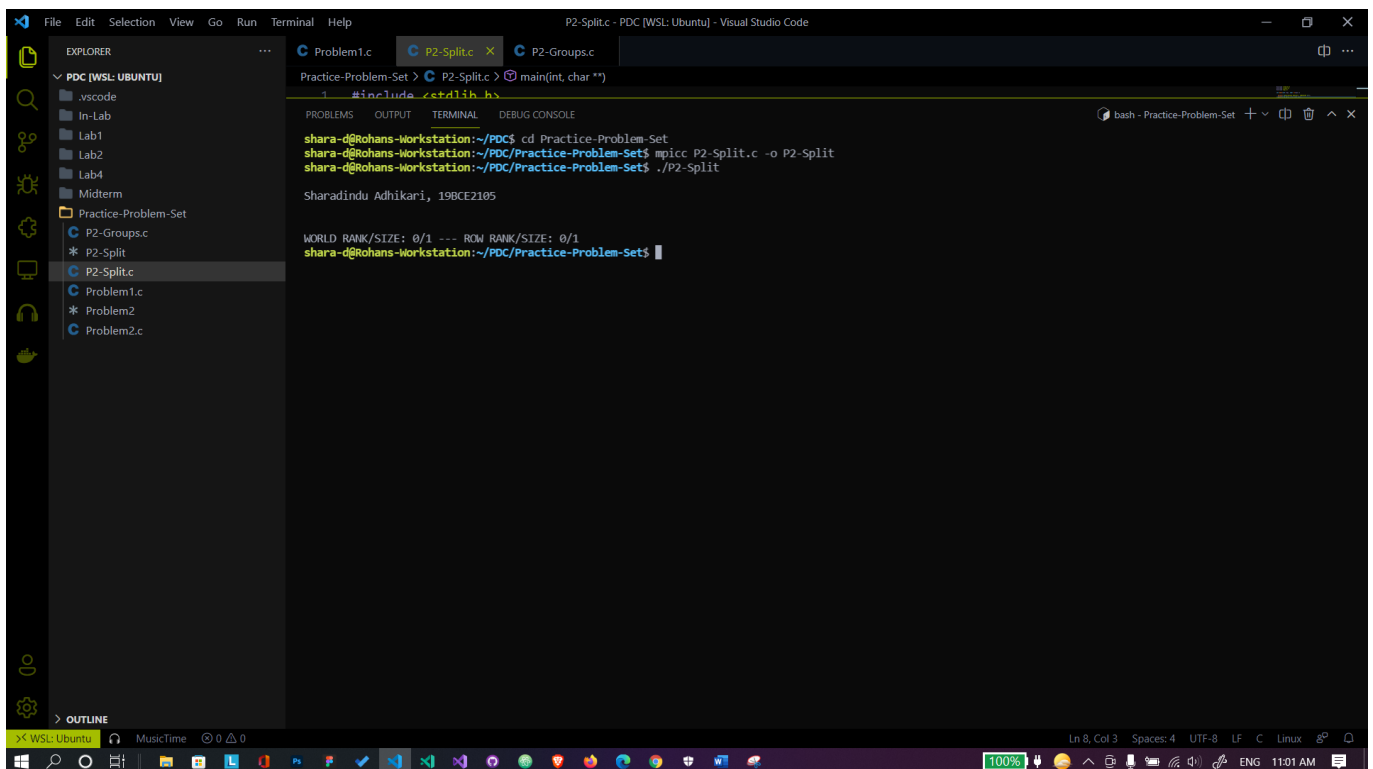
```c
#include <stdlib.h>
#include <stdio.h>
#include <mpi.h>

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

    printf("\nSharadindu Adhikari, 19BCE2105 \n");
    printf("\n \n"); //for some space

    MPI_Init(NULL, NULL);

    // Get the rank and size in the original communicator
    int world_rank, world_size;
    MPI_Comm_rank(MPI_COMM_WORLD, &world_rank);
    MPI_Comm_size(MPI_COMM_WORLD, &world_size);

    int color = world_rank / 4; // Determine color based on row

    // Split the communicator based on the color and use the original rank for ordering
    MPI_Comm row_comm;
    MPI_Comm_split(MPI_COMM_WORLD, color, world_rank, &row_comm);

    int row_rank, row_size;
    MPI_Comm_rank(row_comm, &row_rank);
    MPI_Comm_size(row_comm, &row_size);

    printf("WORLD RANK/SIZE: %d/%d --- ROW RANK/SIZE: %d/%d\n",
        world_rank, world_size, row_rank, row_size);

    MPI_Comm_free(&row_comm);

    MPI_Finalize();
}
```

Output:

```
shara-d@Rohans-Workstation:~/PDC$ cd Practice-Problem-Set
shara-d@Rohans-Workstation:~/PDC/Practice-Problem-Set$ mpicc P2-Split.c -o P2-Split
shara-d@Rohans-Workstation:~/PDC/Practice-Problem-Set$ ./P2-Split

Sharadindu Adhikari, 19BCE2105


WORLD RANK/SIZE: 0/1 --- ROW RANK/SIZE: 0/1
shara-d@Rohans-Workstation:~/PDC/Practice-Problem-Set$
```