MONASH University
Information Technology

GROUP ASSIGNMENT COVER SHEET

Student ID Number	Surname	Given Names
31878822	*Wong*	Mun Kiat
31989403	*Hee*	Zhan Zhynn

* Please include the names of all other group members.

Unit name and code	FIT3143 : Parallel Computing	
Title of assignment	FIT3143 Assignment 2	
Lecturer/tutor	Dr. Vishnu Monn	
Tutorial day and time	Tuesday, 11am - 12pm	**Campus**
Is this an authorised group assignment?	☐ **Yes** ☐ **No**	
Has any part of this assignment been previously submitted as part of another unit/course?	☐ **Yes** ☐ **No**	
Due Date 18/10/2022		**Date submitted** 18/10/2022

All work must be submitted by the due date. If an extension of work is granted this must be specified with the signature of the lecturer/tutor.

Extension granted until (date) .... **Signature of lecturer/tutor** ....

Please note that it is your responsibility to retain copies of your assessments.

Intentional plagiarism or collusion amounts to cheating under Part 7 of the Monash University (Council) Regulations

Plagiarism: Plagiarism means taking and using another person's ideas or manner of expressing them and passing them off as one's own. For example, by failing to give appropriate acknowledgement. The material used can be from any source (staff, students or the internet, published and unpublished works).

Collusion: Collusion means unauthorised collaboration with another person on assessable written, oral or practical work and includes paying another person to complete all or part of the work.

Where there are reasonable grounds for believing that intentional plagiarism or collusion has occurred, this will be reported to the Associate Dean (Education) or delegate, who may disallow the work concerned by prohibiting assessment or refer the matter to the Faculty Discipline Panel for a hearing.

Student Statement:

- I have read the university's Student Academic Integrity Policy and Procedures.
- I understand the consequences of engaging in plagiarism and collusion as described in Part 7 of the Monash University (Council) Regulations http://adm.monash.edu/legal/legislation/statutes
- I have taken proper care to safeguard this work and made all reasonable efforts to ensure it could not be copied.
- No part of this assignment has been previously submitted as part of another unit/course.
- I acknowledge and agree that the assessor of this assignment may for the purposes of assessment, reproduce the assignment and:
    i. provide to another member of faculty and any external marker; and/or
    ii. submit it to a text matching software; and/or
    iii. submit it to a text matching software which may then retain a copy of the assignment on its database for the purpose of future plagiarism checking.
- I certify that I have not plagiarised the work of others or participated in unauthorised collaboration when preparing this assignment.

Signature .... *Date*...

* delete (iii) if not applicable

Signature WONG MUN KIAT Date: 16/10/2022 Signature WONG MUN KIAT Date: 16/10/2022

Signature HEE ZHAN ZHYNN Date: 16/10/2022 Signature HEE ZHAN ZHYNN Date: 16/10/2022

MONASH University

FIT3143 Semester 2, 2022
Assignment 2 - Report

Team Name (or Number): Lab 06 - Team 03

Student email address	Student First Name	Student Last Name	Contribution %	Contribution details*
mwon0044@student.monash.edu	Mun Kiat	Wong	50%	Base Station, Balloon Seismic Sensor, Report
zhee0035@student.monash.edu	Zhan Zhynn	Hee	50%	Seafloor Sensor Node, Balloon Seismic Sensor, Report

**Your contribution details include the report, code, or both.*

Note: Please refer to Assignment specifications, FAQ and marking guidelines for details to be included in the following sections of this report.

Include the word count here (for Sections A to C): 1552 words

A. Methodology

1. Seafloor Seismic Sensor Nodes

The sensor node is a MPI virtual topology where each node can exchange data with the immediate adjacent nodes as shown in Figure 1.1. Each node generates random values within a specified range for the magnitude, longitude, latitude and depth as shown in Figure 1.2.

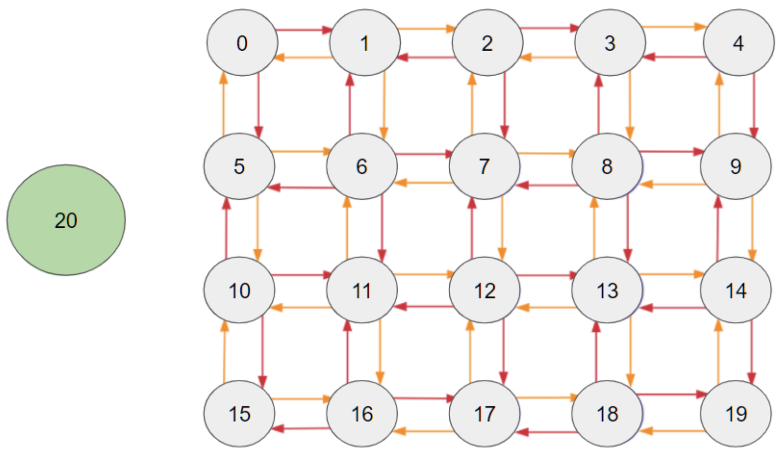

Figure 1.1: Example architecture of seafloor seismic sensor nodes

```
Time:18-10-2022 21:53:39, rank: 0, randomVal: 7, lat: -14, long: 167, depth: 6
```

Figure 1.2: Simulated value by a sensor node

When the magnitude simulated is above threshold, the node sends a request to all adjacent nodes by using the REQUEST tag and the content sent is the node's rank. Each node only sends its simulated values when it receives the REQUEST tag instead of broadcasting to all adjacent nodes.

When a node receives the REQUEST tag, it stores the received value (the requester node's rank). The node then uses the REPLY tag and sends the requested values back to the requester node. It uses the requester node's rank stored earlier as the sending destination. The implementation is as shown in figure 1.3 .

Figure 1.3: Seafloor Seismic Sensor Nodes Send-Receive Process

When the requester receives the requested values back from the adjacent nodes, it then compares the magnitude and distance to determine if an event occurred. If there are more than two neighbouring events, it sends a report back to the base station. An example of the received values is as follow where H rank indicates the node with values above threshold:

```
H rank: 1, received: 5, lat: -15, long: 166, from 0
```

Figure 1.4 : Received Value

For sensor nodes with values below the threshold, it does not send messages to the adjacent nodes, thus, it does not explicitly receive any as well. However, since the adjacent nodes might send a request, it needs to be aware of the sending request. This is done using MPI_Iprobe where it probes whether there's a message being sent to the node, this helps to prevent any deadlock.

Note: Sensor nodes generate integer values only instead of floats due to issues of sending and receiving float values in MPI processes. We have informed the lab tutors about it.

2. Balloon Seismic Sensor

The Balloon Seismic Sensor is part of the WSN where it only communicates with the base station. This part of the WSN consists of a shared global array that can be accessed by the base station. The array has a fixed size (size 5) in which the values are produced every cycle (1s). Values are generated through the number generators and localtime() for time. It is a first in first out array where new values are stored by using mod maths to get the index of the array that should be replaced first. To signal the balloon seismic sensor to terminate, we first initialise a global variable balloon_terminate to 0, then the base station will update the variable to an integer 1 to signal termination. This way, we do not need to send a message to the sensor as the sensor only needs to read from the global array, hence there is no race condition as well.

```
while no terminate:
  value = generate values for balloon readings
  globalarray[0...3] = earthquake readings (mag, lat, long, depth)
  globalarray[4...9] = time (year, month, day, hour, min, sec)
...
```

Figure 2.1 : Pseudocode for balloon readings

```
balloon array:
[Time: 2022-10-18 15:54:1, Mag: 4, Lat: -16, Long: 164, Depth(km): 3]
[Time: 2022-10-18 15:54:2, Mag: 5, Lat: -15, Long: 165, Depth(km): 4]
[Time: 2022-10-18 15:54:3, Mag: 6, Lat: -15, Long: 166, Depth(km): 5]
[Time: 2022-10-18 15:53:59, Mag: 7, Lat: -14, Long: 167, Depth(km): 6]
[Time: 2022-10-18 15:54:0, Mag: 8, Lat: -14, Long: 167, Depth(km): 6]
```

Figure 2.2 : Sample output for balloon

To build an efficient Balloon Seismic Sensor, the team utilised the **POSIX Threads (pthreads)** which is a parallel execution model (can be seen in Figure 2.3). The base station creates a thread using the **pthread_create** and runs the **ProcessFunc** which produces the readings for the Balloon Seismic Sensor. The thread runs in a continuous loop until a termination message is received from the base station.

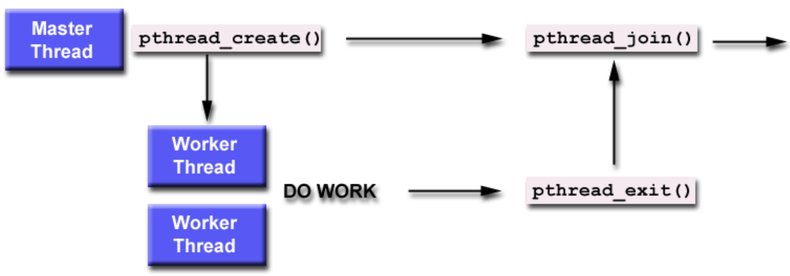

Figure 2.3 : pthreads

3. Base station

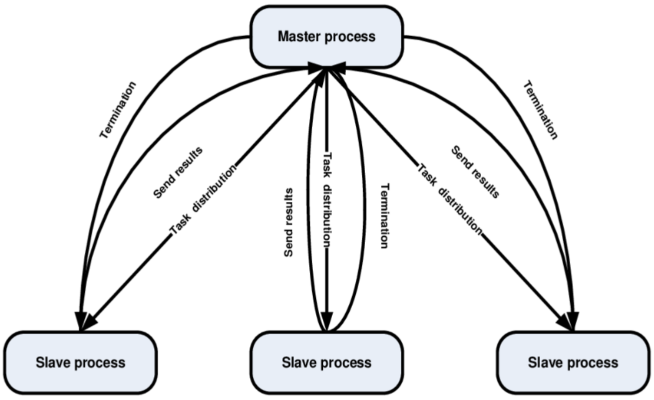

Figure 3.1 : Master / Slave

As seen in Figure 3.1, the base station acts as a master which receives reports from the sensor nodes. . It represents a node which gathers data from all sensor nodes within the WSN. Working alongside the Sensor Nodes and Balloon Sensors, it uses a simple master and slave approach to delegate tasks in the WSN.

To allow the base station to work effectively, we used Message Passing Interface (MPI) which is an interface that allows programs to send messages from one process to another. The base station is then built in a parallel approach which improves efficiency. As the WSN sends data to the base station, the base station will call MPI_Recv which is a blocking receiver for messages. In Figure 3.2, MPI_Recv(s) are used to receive the messages to the initial buffer and status from the given communicator, tag and so on. If tagged DETECTED, it receives the "receive_ar" array containing the necessary information on the detected earthquake readings along with the time. It then compares values with the shared global array. If matched, it will be reported as a Conclusive Alert, otherwise, an Inconclusive Alert. It is important to note that the base station is only reading from the global array, hence we do not need a mutex lock as there will not be any race conditions.

```
MPI_Recv(buffer, 256, MPI_CHAR, i, MPI_ANY_TAG, master_comm, &status);

if status has DETECTED_TAG:
  MPI_Recv(receive_array, 11, MPI_INT, i, DETECTED_TAG, master_comm, &status);

  if (compare receive_array with shared global array):
    match : Conclusive Alert
    no match : Inconclusive Alert
...
```

Figure 3.2 : Pseudocode of MPI_Recv

The “Base Station Log” contains the detected earthquake readings along with the key performance metrics such as (1) simulation time, (2) number of alerts detected, (3) number of messages/events with senders’ adjacency information/addresses, (4) total number of messages, etc. The base station log also includes the Received Nodes that are detected along with their location and magnitude. The **“log.txt”** is created using the fopen(), fprintf(), fclose(), etc. functions which are all for the creation, opening, writing, appending or even closing of a file. A sample of the log file can be seen in Figure 3.3.

Lastly, the base station has a termination feature where if a user inputs the SENTINEL value of “-1” into the txt file, it will read the txt file to check for its termination status, if true, it sends the termination message to both sensor nodes & balloon sensor for the termination of program.

```
========================================================================
BASE STATION LOG

Received from Node 2
Simulation: 2022-10-18 21:47:58,Magnitude: 6, Latitude: -15, Longitude: 166, Depth(km): 5
Type of Alert: Conclusive Alert, Number of Neighbours: 2, Total Messages: 3

Received from Node 6
Simulation: 2022-10-18 21:47:58,Magnitude: 5, Latitude: -15, Longitude: 166, Depth(km): 5
Type of Alert: Conclusive Alert, Number of Neighbours: 2, Total Messages: 7

Received from Node 7
Simulation: 2022-10-18 21:47:58,Magnitude: 6, Latitude: -14, Longitude: 166, Depth(km): 5
Type of Alert: Conclusive Alert, Number of Neighbours: 3, Total Messages: 8

Received from Node 8
Simulation: 2022-10-18 21:47:58,Magnitude: 8, Latitude: -14, Longitude: 167, Depth(km): 6
Type of Alert: Conclusive Alert, Number of Neighbours: 2, Total Messages: 9

Received from Node 10
Simulation: 2022-10-18 21:47:58,Magnitude: 4, Latitude: -15, Longitude: 165, Depth(km): 4
Type of Alert: Conclusive Alert, Number of Neighbours: 2, Total Messages: 11

Received from Node 11
Simulation: 2022-10-18 21:47:58,Magnitude: 6, Latitude: -15, Longitude: 166, Depth(km): 5
Type of Alert: Conclusive Alert, Number of Neighbours: 2, Total Messages: 12

Total Number of Alerts: 6
```

Figure 3.3: base station log.txt

B. Results Tabulation

Note : All results consist of 3 iterations (ran 3 times).

No. of runs	Grid Size	Total Earthquakes	Total Messages	Summary
3	2 x 2	0	0	Ran on **local computer**. No earthquake for all 3 runs
3	4 x 4	29	42	Ran on **local computer**. 29 earthquakes detected along with 42 messages exchanged with base station at 3 runs
3	9 x 3	38	76	Ran on **CAAS**. 38 earthquakes detected along with 76 messages exchanged with base station at 3 runs

Note : All terminal output & base station log screenshots can be seen in the figures below. Screenshots are not complete as some logs contain too much information.

1. Number of Nodes = 5 , Grid Size = 2 x 2

```
mpirun -oversubscribe -np 5 main_out

Root Rank: 0. Comm Size: 4: Grid Dimension = [2 x 2]
Mon Oct 17 21:44:00 2022
rank: 0, randomVal: 3, lat: -15, long: 165, depth: 4
Mon Oct 17 21:44:00 2022
rank: 1, randomVal: 5, lat: -15, long: 166, depth: 5
Mon Oct 17 21:44:00 2022
rank: 2, randomVal: 7, lat: -14, long: 167, depth: 6
Mon Oct 17 21:44:00 2022
rank: 3, randomVal: 1, lat: -16, long: 164, depth: 3
H rank: 0, received: 5, lat: -15, long: 166, from 1
H rank: 0, received: 7, lat: -14, long: 167, from 2
H rank: 1, received: 3, lat: -15, long: 165, from 0
H rank: 2, received: 3, lat: -15, long: 165, from 0
H rank: 2, received: 1, lat: -16, long: 164, from 3
H rank: 1, received: 1, lat: -16, long: 164, from 3
No earthquake at 0
No earthquake at 1
No earthquake at 2
No earthquake at 3
```

Figure 4.1 : np5 terminal output 1

```
Root Rank: 0. Comm Size: 4: Grid Dimension = [2 x 2]
Mon Oct 17 21:44:10 2022
rank: 0, randomVal: 5, lat: -15, long: 166, depth: 5
Mon Oct 17 21:44:10 2022
rank: 1, randomVal: 7, lat: -14, long: 167, depth: 6
Mon Oct 17 21:44:10 2022
rank: 2, randomVal: 1, lat: -16, long: 164, depth: 3
Mon Oct 17 21:44:10 2022
rank: 3, randomVal: 2, lat: -16, long: 164, depth: 3
H rank: 1, received: 5, lat: -15, long: 166, from 0
H rank: 0, received: 7, lat: -14, long: 167, from 1
H rank: 0, received: 1, lat: -16, long: 164, from 2
H rank: 1, received: 2, lat: -16, long: 164, from 3
No earthquake at 0
No earthquake at 1
No earthquake at 2
No earthquake at 3
```

Figure 4.2 : np5 terminal output 2

```
Root Rank: 0. Comm Size: 4: Grid Dimension = [2 x 2]
Mon Oct 17 21:44:20 2022
rank: 0, randomVal: 7, lat: -14, long: 167, depth: 6
Mon Oct 17 21:44:20 2022
rank: 1, randomVal: 1, lat: -16, long: 164, depth: 3
Mon Oct 17 21:44:20 2022
rank: 2, randomVal: 2, lat: -16, long: 164, depth: 3
Mon Oct 17 21:44:20 2022
rank: 3, randomVal: 4, lat: -15, long: 165, depth: 4
H rank: 0, received: 2, lat: -16, long: 164, from 2
H rank: 0, received: 1, lat: -16, long: 164, from 1
H rank: 3, received: 1, lat: -16, long: 164, from 1
H rank: 3, received: 2, lat: -16, long: 164, from 2
No earthquake at 0
No earthquake at 1
No earthquake at 2
No earthquake at 3
```

Figure 4.3 : np5 terminal output 3

2. Number of Nodes = 17 , Grid Size = 4 x 4

```
mpirun -oversubscribe -np 17 main_out

Root Rank: 0. Comm Size: 16: Grid Dimension = [4 x 4]
Mon Oct 17 21:38:05 2022
rank: 0, randomVal: 5, lat: -15, long: 166, depth: 5
Mon Oct 17 21:38:05 2022
rank: 1, randomVal: 7, lat: -14, long: 167, depth: 6
Mon Oct 17 21:38:05 2022
rank: 2, randomVal: 1, lat: -16, long: 164, depth: 3
Mon Oct 17 21:38:05 2022
rank: 3, randomVal: 3, lat: -16, long: 165, depth: 4
Mon Oct 17 21:38:05 2022
rank: 4, randomVal: 4, lat: -15, long: 165, depth: 4
Mon Oct 17 21:38:05 2022
rank: 5, randomVal: 6, lat: -14, long: 166, depth: 5
Mon Oct 17 21:38:05 2022
rank: 6, randomVal: 8, lat: -14, long: 167, depth: 6
Mon Oct 17 21:38:05 2022
rank: 7, randomVal: 2, lat: -16, long: 164, depth: 3
Mon Oct 17 21:38:05 2022
rank: 8, randomVal: 3, lat: -15, long: 165, depth: 4
Mon Oct 17 21:38:05 2022
rank: 9, randomVal: 5, lat: -15, long: 166, depth: 5
Mon Oct 17 21:38:05 2022
```

Figure 5.1 : np17 terminal output 1

```
H rank: 4, received: 5, lat: -15, long: 166, from 0
H rank: 3, received: 1, lat: -16, long: 164, from 2
H rank: 6, received: 2, lat: -16, long: 164, from 7
H rank: 15, received: 1, lat: -16, long: 164, from 11
H rank: 1, received: 1, lat: -16, long: 164, from 2
H rank: 10, received: 1, lat: -16, long: 164, from 11
H rank: 3, received: 2, lat: -16, long: 164, from 7
H rank: 6, received: 1, lat: -16, long: 164, from 2
Earthquake verified at 0
Earthquake verified at 1
No earthquake at 2
No earthquake at 3
Earthquake verified at 4
Earthquake verified at 5
Earthquake verified at 6
No earthquake at 7
Earthquake verified at 8
Earthquake verified at 9
Earthquake verified at 10
No earthquake at 11
Earthquake verified at 12
Earthquake verified at 13
Earthquake verified at 14
No earthquake at 15
```

Figure 5.2 : np17 terminal output 2

```
========================================================================
BASE STATION LOG

Received from Node 0
Simulation: Date Mon Oct 17 21:38:05 2022
Magnitude: 5, Latitude: -15, Longitude: 166, Depth(km): 5
Type of Alert: Conclusive Alert, Number of Neighbours: 2, Total Messages: 1

Received from Node 1
Simulation: Date Mon Oct 17 21:38:05 2022
Magnitude: 7, Latitude: -14, Longitude: 167, Depth(km): 6
Type of Alert: Conclusive Alert, Number of Neighbours: 2, Total Messages: 2

Received from Node 4
Simulation: Date Mon Oct 17 21:38:05 2022
Magnitude: 4, Latitude: -15, Longitude: 165, Depth(km): 4
Type of Alert: Conclusive Alert, Number of Neighbours: 3, Total Messages: 5

Received from Node 5
Simulation: Date Mon Oct 17 21:38:05 2022
Magnitude: 6, Latitude: -14, Longitude: 166, Depth(km): 5
Type of Alert: Conclusive Alert, Number of Neighbours: 4, Total Messages: 6

Received from Node 6
Simulation: Date Mon Oct 17 21:38:05 2022
Magnitude: 8, Latitude: -14, Longitude: 167, Depth(km): 6
Type of Alert: Conclusive Alert, Number of Neighbours: 2, Total Messages: 7

Received from Node 8
Simulation: Date Mon Oct 17 21:38:05 2022
Magnitude: 3, Latitude: -15, Longitude: 165, Depth(km): 4
Type of Alert: Conclusive Alert, Number of Neighbours: 3, Total Messages: 9
```

Figure 5.3: np17 base station log (run 1)

```
Received from Node 9
Simulation: Date Mon Oct 17 21:38:05 2022
Magnitude: 5, Latitude: -15, Longitude: 166, Depth(km): 5
Type of Alert: Conclusive Alert, Number of Neighbours: 4, Total Messages: 10

Received from Node 10
Simulation: Date Mon Oct 17 21:38:05 2022
Magnitude: 7, Latitude: -14, Longitude: 167, Depth(km): 6
Type of Alert: Conclusive Alert, Number of Neighbours: 3, Total Messages: 11

Received from Node 12
Simulation: Date Mon Oct 17 21:38:05 2022
Magnitude: 3, Latitude: -16, Longitude: 165, Depth(km): 4
Type of Alert: Conclusive Alert, Number of Neighbours: 2, Total Messages: 13

Received from Node 13
Simulation: Date Mon Oct 17 21:38:05 2022
Magnitude: 4, Latitude: -15, Longitude: 165, Depth(km): 4
Type of Alert: Conclusive Alert, Number of Neighbours: 3, Total Messages: 14

Received from Node 14
Simulation: Date Mon Oct 17 21:38:05 2022
Magnitude: 6, Latitude: -14, Longitude: 166, Depth(km): 5
Type of Alert: Conclusive Alert, Number of Neighbours: 3, Total Messages: 15

Total Number of Alerts: 11
```

Figure 5.4: np17 base station log (run 1)

```
========================================================================
BASE STATION LOG

Received from Node 2
Simulation: Date Mon Oct 17 21:38:39 2022
Magnitude: 5, Latitude: -15, Longitude: 166, Depth(km): 5
Type of Alert: Conclusive Alert, Number of Neighbours: 3, Total Messages: 3

Received from Node 3
Simulation: Date Mon Oct 17 21:38:39 2022
Magnitude: 7, Latitude: -14, Longitude: 167, Depth(km): 6
Type of Alert: Conclusive Alert, Number of Neighbours: 2, Total Messages: 4

Received from Node 6
Simulation: Date Mon Oct 17 21:38:39 2022
Magnitude: 4, Latitude: -15, Longitude: 165, Depth(km): 4
Type of Alert: Conclusive Alert, Number of Neighbours: 3, Total Messages: 7

Received from Node 7
Simulation: Date Mon Oct 17 21:38:39 2022
Magnitude: 6, Latitude: -14, Longitude: 166, Depth(km): 5
Type of Alert: Conclusive Alert, Number of Neighbours: 3, Total Messages: 8

Received from Node 10
Simulation: Date Mon Oct 17 21:38:39 2022
Magnitude: 3, Latitude: -15, Longitude: 165, Depth(km): 4
Type of Alert: Conclusive Alert, Number of Neighbours: 2, Total Messages: 11

Received from Node 11
Simulation: Date Mon Oct 17 21:38:39 2022
Magnitude: 5, Latitude: -15, Longitude: 166, Depth(km): 5
Type of Alert: Conclusive Alert, Number of Neighbours: 3, Total Messages: 12

Total Number of Alerts: 6
```

Figure 5.5 : np17 base station log (run 2)

```
Received from Node 9
Simulation: Date Mon Oct 17 21:39:13 2022
Magnitude: 6, Latitude: -14, Longitude: 166, Depth(km): 5
Type of Alert: Conclusive Alert, Number of Neighbours: 4, Total Messages: 10

Received from Node 10
Simulation: Date Mon Oct 17 21:39:13 2022
Magnitude: 8, Latitude: -14, Longitude: 167, Depth(km): 6
Type of Alert: Conclusive Alert, Number of Neighbours: 2, Total Messages: 11

Received from Node 12
Simulation: Date Mon Oct 17 21:39:13 2022
Magnitude: 3, Latitude: -15, Longitude: 165, Depth(km): 4
Type of Alert: Conclusive Alert, Number of Neighbours: 2, Total Messages: 13

Received from Node 13
Simulation: Date Mon Oct 17 21:39:13 2022
Magnitude: 5, Latitude: -15, Longitude: 166, Depth(km): 5
Type of Alert: Conclusive Alert, Number of Neighbours: 3, Total Messages: 14

Received from Node 14
Simulation: Date Mon Oct 17 21:39:13 2022
Magnitude: 7, Latitude: -14, Longitude: 167, Depth(km): 6
Type of Alert: Conclusive Alert, Number of Neighbours: 2, Total Messages: 15

Total Number of Alerts: 10
```

Figure 5.6 : np17 base station log (run 3)

3. [CAAS] Number of Nodes = 28, Grid Size = 9 x 3

```
[zhee0035@student-caas-headnode FIT3143_Ass2_CAAS]$ squeue
             JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
              2785      defq  mpi_job   jpha23 PD       0:00      8 (PartitionConfig)
              4302      defq driver_j nemm0001 PD       0:00     14 (PartitionConfig)
              4303      defq driver_j nemm0001 PD       0:00     14 (PartitionConfig)
              4333      defq driver_j nemm0001 PD       0:00     16 (PartitionConfig)
              4373      defq driver_j nemm0001 PD       0:00     13 (PartitionConfig)
              4377      defq driver_j nemm0001 PD       0:00      6 (PartitionConfig)
              4378      defq driver_j nemm0001 PD       0:00      1 (PartitionConfig)
              4379      defq driver_j nemm0001 PD       0:00      4 (PartitionConfig)
              4400      defq Assignme ymaa0058 PD       0:00      8 (PartitionConfig)
              4401      defq Assignme ymaa0058 PD       0:00      8 (PartitionConfig)
              4555      defq caas_job fbon0004 PD       0:00     11 (PartitionConfig)
              4556      defq caas_job fbon0004 PD       0:00     11 (PartitionConfig)
              4924      defq ass2_job zhee0035  R       2:36      2 student-caas-n[01-02]
              4925      defq main_job msim0015  R       1:35      3 student-caas-n[03-05]
[zhee0035@student-caas-headnode FIT3143_Ass2_CAAS]$
```

Figure 6.1: Job queue at CAAS

```
mpicc -c -Wall -Wextra -std=c99 -pedantic -g main.c
mpicc -L pcg-c-0.94/src/ main.o slave.o master.o -o main_out -lm
No earthquake at 0
No earthquake at 1
Earthquake verified at 2
No earthquake at 3
No earthquake at 4
No earthquake at 5
Earthquake verified at 6
Earthquake verified at 7
Earthquake verified at 8
No earthquake at 9
Earthquake verified at 10
Earthquake verified at 11
No earthquake at 12
No earthquake at 13
No earthquake at 14
Earthquake verified at 15
Earthquake verified at 16
Earthquake verified at 17
No earthquake at 18
```

Figure 6.2: sample CAAS terminal output 1

```
Earthquake verified at 12
Earthquake verified at 13
No earthquake at 14
Earthquake verified at 15
Earthquake verified at 16
Earthquake verified at 17
No earthquake at 18
No earthquake at 19
No earthquake at 20
Earthquake verified at 21
Earthquake verified at 22
No earthquake at 23
Earthquake verified at 24
Earthquake verified at 25
No earthquake at 26

Root Rank: 0. Comm Size: 27: Grid Dimension = [9 x 3]

Root Rank: 0. Comm Size: 27: Grid Dimension = [9 x 3]

Root Rank: 0. Comm Size: 27: Grid Dimension = [9 x 3]
```

Figure 6.3 : sample CAAS terminal output 2

```
========================================================================
BASE STATION LOG

Received from Node 2
Simulation: Date Tue Oct 18 03:27:24 2022
Magnitude: 5, Latitude: -15, Longitude: 166, Depth(km): 5
Type of Alert: Conclusive Alert, Number of Neighbours: 2, Total Messages: 3

Received from Node 6
Simulation: Date Tue Oct 18 03:27:24 2022
Magnitude: 4, Latitude: -15, Longitude: 165, Depth(km): 4
Type of Alert: Conclusive Alert, Number of Neighbours: 2, Total Messages: 7

Received from Node 7
Simulation: Date Tue Oct 18 03:27:24 2022
Magnitude: 6, Latitude: -14, Longitude: 166, Depth(km): 5
Type of Alert: Conclusive Alert, Number of Neighbours: 3, Total Messages: 8

Received from Node 8
Simulation: Date Tue Oct 18 03:27:24 2022
Magnitude: 8, Latitude: -14, Longitude: 167, Depth(km): 6
Type of Alert: Conclusive Alert, Number of Neighbours: 2, Total Messages: 9

Received from Node 10
Simulation: Date Tue Oct 18 03:27:24 2022
Magnitude: 4, Latitude: -15, Longitude: 165, Depth(km): 4
Type of Alert: Conclusive Alert, Number of Neighbours: 2, Total Messages: 11
```

Figure 6.4: CAAS base station log (run 1)

```
Received from Node 17
Simulation: Date Tue Oct 18 03:27:24 2022
Magnitude: 8, Latitude: -14, Longitude: 167, Depth(km): 6
Type of Alert: Conclusive Alert, Number of Neighbours: 2, Total Messages: 18

Received from Node 19
Simulation: Date Tue Oct 18 03:27:24 2022
Magnitude: 3, Latitude: -15, Longitude: 165, Depth(km): 4
Type of Alert: Conclusive Alert, Number of Neighbours: 2, Total Messages: 20

Received from Node 20
Simulation: Date Tue Oct 18 03:27:24 2022
Magnitude: 5, Latitude: -15, Longitude: 166, Depth(km): 5
Type of Alert: Conclusive Alert, Number of Neighbours: 3, Total Messages: 21

Received from Node 24
Simulation: Date Tue Oct 18 03:27:24 2022
Magnitude: 4, Latitude: -15, Longitude: 165, Depth(km): 4
Type of Alert: Conclusive Alert, Number of Neighbours: 2, Total Messages: 25

Received from Node 25
Simulation: Date Tue Oct 18 03:27:24 2022
Magnitude: 6, Latitude: -14, Longitude: 166, Depth(km): 5
Type of Alert: Conclusive Alert, Number of Neighbours: 2, Total Messages: 26

Total Number of Alerts: 13
```

Figure 6.5 : CAAS base station log (run 1)

```
==========================================================================
BASE STATION LOG

Received from Node 1
Simulation: Date Tue Oct 18 03:28:20 2022
Magnitude: 7, Latitude: -14, Longitude: 167, Depth(km): 6
Type of Alert: Conclusive Alert, Number of Neighbours: 2, Total Messages: 2

Received from Node 4
Simulation: Date Tue Oct 18 03:28:20 2022
Magnitude: 4, Latitude: -15, Longitude: 165, Depth(km): 4
Type of Alert: Conclusive Alert, Number of Neighbours: 2, Total Messages: 5

Received from Node 5
Simulation: Date Tue Oct 18 03:28:20 2022
Magnitude: 6, Latitude: -14, Longitude: 166, Depth(km): 5
Type of Alert: Conclusive Alert, Number of Neighbours: 2, Total Messages: 6

Received from Node 9
Simulation: Date Tue Oct 18 03:28:20 2022
Magnitude: 5, Latitude: -15, Longitude: 166, Depth(km): 5
Type of Alert: Conclusive Alert, Number of Neighbours: 2, Total Messages: 10

Received from Node 10
Simulation: Date Tue Oct 18 03:28:20 2022
Magnitude: 7, Latitude: -14, Longitude: 167, Depth(km): 6
Type of Alert: Conclusive Alert, Number of Neighbours: 2, Total Messages: 11
```

Figure 6.6 : CAAS base station log (run 2)

```
Received from Node 14
Simulation: Date Tue Oct 18 03:28:20 2022
Magnitude: 6, Latitude: -14, Longitude: 166, Depth(km): 5
Type of Alert: Conclusive Alert, Number of Neighbours: 2, Total Messages: 15

Received from Node 18
Simulation: Date Tue Oct 18 03:28:20 2022
Magnitude: 5, Latitude: -15, Longitude: 166, Depth(km): 5
Type of Alert: Conclusive Alert, Number of Neighbours: 2, Total Messages: 19

Received from Node 19
Simulation: Date Tue Oct 18 03:28:20 2022
Magnitude: 7, Latitude: -14, Longitude: 167, Depth(km): 6
Type of Alert: Conclusive Alert, Number of Neighbours: 2, Total Messages: 20

Received from Node 22
Simulation: Date Tue Oct 18 03:28:20 2022
Magnitude: 4, Latitude: -15, Longitude: 165, Depth(km): 4
Type of Alert: Conclusive Alert, Number of Neighbours: 2, Total Messages: 23

Received from Node 23
Simulation: Date Tue Oct 18 03:28:20 2022
Magnitude: 6, Latitude: -14, Longitude: 166, Depth(km): 5
Type of Alert: Conclusive Alert, Number of Neighbours: 2, Total Messages: 24

Total Number of Alerts: 11
```

Figure 6.7 : CAAS base station log (run 2)

```
==========================================================================
BASE STATION LOG

Received from Node 3
Simulation: Date Tue Oct 18 03:29:16 2022
Magnitude: 6, Latitude: -14, Longitude: 166, Depth(km): 5
Type of Alert: Conclusive Alert, Number of Neighbours: 2, Total Messages: 4

Received from Node 4
Simulation: Date Tue Oct 18 03:29:16 2022
Magnitude: 8, Latitude: -14, Longitude: 167, Depth(km): 6
Type of Alert: Conclusive Alert, Number of Neighbours: 2, Total Messages: 5

Received from Node 6
Simulation: Date Tue Oct 18 03:29:16 2022
Magnitude: 3, Latitude: -15, Longitude: 165, Depth(km): 4
Type of Alert: Conclusive Alert, Number of Neighbours: 2, Total Messages: 7

Received from Node 7
Simulation: Date Tue Oct 18 03:29:16 2022
Magnitude: 5, Latitude: -15, Longitude: 166, Depth(km): 5
Type of Alert: Conclusive Alert, Number of Neighbours: 3, Total Messages: 8

Received from Node 8
Simulation: Date Tue Oct 18 03:29:16 2022
Magnitude: 7, Latitude: -14, Longitude: 167, Depth(km): 6
Type of Alert: Conclusive Alert, Number of Neighbours: 2, Total Messages: 9
```

Figure 6.8 : CAAS base station log (run 3)

```
Received from Node 17
Simulation: Date Tue Oct 18 03:29:16 2022
Magnitude: 7, Latitude: -14, Longitude: 167, Depth(km): 6
Type of Alert: Conclusive Alert, Number of Neighbours: 2, Total Messages: 18

Received from Node 21
Simulation: Date Tue Oct 18 03:29:16 2022
Magnitude: 6, Latitude: -15, Longitude: 166, Depth(km): 5
Type of Alert: Conclusive Alert, Number of Neighbours: 3, Total Messages: 22

Received from Node 22
Simulation: Date Tue Oct 18 03:29:16 2022
Magnitude: 8, Latitude: -14, Longitude: 167, Depth(km): 6
Type of Alert: Conclusive Alert, Number of Neighbours: 2, Total Messages: 23

Received from Node 24
Simulation: Date Tue Oct 18 03:29:16 2022
Magnitude: 3, Latitude: -16, Longitude: 165, Depth(km): 4
Type of Alert: Conclusive Alert, Number of Neighbours: 2, Total Messages: 25

Received from Node 25
Simulation: Date Tue Oct 18 03:29:16 2022
Magnitude: 5, Latitude: -15, Longitude: 166, Depth(km): 5
Type of Alert: Conclusive Alert, Number of Neighbours: 3, Total Messages: 26

Total Number of Alerts: 14
```

Figure 6.9 : CAAS base station log (run 3)

C. Analysis & Discussion

Hypothesis:
Assume each node is an mpi_process, if the number of nodes exceeds the CPU processors, we will be bottlenecked as now each CPU processor will take turns to process the mpi_process.

Inference:
The lower the number of nodes, the lower the overall communication time between nodes and the base station.

Based on the results tabulated in Section B, we can see that:

Type	Grid Size	Total Run Time (sec)
Single Computer	2 x 2	**21.03s** - (8.004 + 6.020 + 7.006)
Single Computer	4 x 4	**94.10s** - (32.013 + 31.045 + 31.045)
Single Computer	9 x 3	**160.15s** - (54.008 + 53.088 + 53.054)
CAAS	9 x 3	**138.22s** - (47.063 + 45.131 + 46.030)

Table 7.1 : Analysis of Results

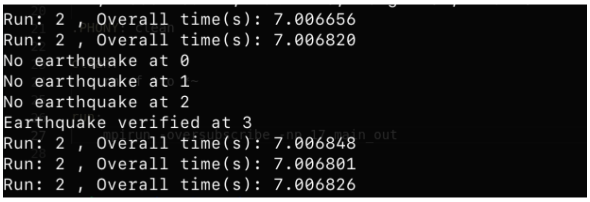

Figure 7.1 : Sample Run Time Display (2x2)

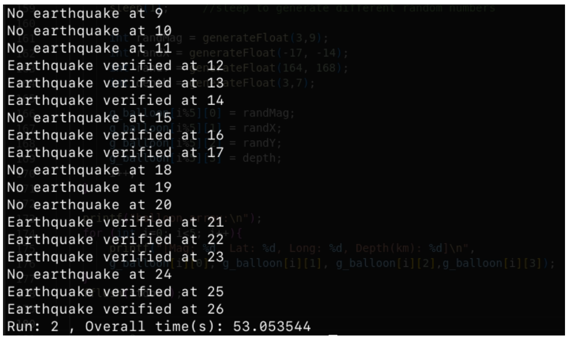

Figure 7.2 : Sample Run Time Display (9x3)

As seen in Table 7.1, the Overall Time of the Single Computer Type increases as the size of the grid increases. This is because a bigger grid size has a larger communication overhead along with higher frequency of blocking and receiving of messages that all adds to the communication time. However, by comparing the Single Computer and CAAS on the grid size of 9x3, we can see that CAAS performs relatively better (22 seconds faster). This is because it has a higher number of cores (16 CPU cores) to handle each mpi process across a cluster which results in a higher level of performance.

The team implemented a rather effective approach on message passing where messages are only exchanged when requested rather than broadcasting to all. This allows faster communication time which significantly reduces the message passing load. For example, instead of overwhelming the base station with all reports, the nodes compare values with adjacent nodes and ensure it has > 1 adjacent nodes detecting events before sending them to the base station. This reduces communication messages between sensor nodes and base stations along with the processing that needs to be done on both ends.

Known Issues & Cause:
If the number of sensor nodes increases, there is a known issue causing the program to go into a deadlock state when a message is sent between nodes that are above and below the threshold. Since each node is an MPI process, it is possible that the nodes below the threshold finish executing MPI_Iprobe and MPI_Recv even before the nodes above the threshold call MPI_Send. This becomes an issue in blocking communication where one side sends but the other side does not receive.

Possible Solution:
To solve this problem, we explicitly force the nodes below the threshold to sleep for a definite period of time. This allows time for nodes above the threshold to call an MPI_Send, then the nodes below the threshold could execute the MPI_Recv.

D. References

Kiepert, J., & Loo, S. M. (2012). A unified wireless sensor network framework. 2012 IEEE International Systems Conference SysCon 2012. doi:10.1109/syscon.2012.6189450 https://ieeexplore.ieee.org/abstract/document/6189450

Ismail, M. A. (n.d.). Parallel genetic algorithms (PGAs): master slave paradigm approach using MPI. E-Tech 2004. doi:10.1109/etech.2004.1353848 https://ieeexplore.ieee.org/abstract/document/1353848

Declaration:

I declare that this assignment report and the submitted code represent work within my team. I have not copied from any other teams' work or from any other source except where due acknowledgment is made explicitly in the report and code, nor has any part of this submission been written for me by another person outside of my team.

Signature of student 1:__________WONG MUN KIAT_____________

Signature of student 2:__________HEE ZHAN ZHYNN_____________