Duration: *50 minutes* (11:10am–12:00pm)

Aids Allowed: *One <u>non</u>-programmable calculator.*

Student Number: |⎵_⎵_⎵_⎵_⎵_⎵_⎵_⎵_⎵_⎵|

Last Name: _____

First Name: _____

Tutorial Section: (circle one)	SS 1083 (Bijan)	SS 1074 (Kiran)	SS 2111 (Hojjat)	SS 2130 (Jerry)	LM 155 (Mike)

---

*Do **not** turn this page until you have received the signal to start.*
(In the meantime, please fill out the identification section above,
and read the instructions below *carefully*.)

This test consists of 3 questions on 5 pages (including this one). *When you receive the signal to start, please make sure that your copy of the test is complete.* Answer each question directly on the test paper, in the space provided, and *use the reverse side of the pages for rough work.* (If you need more space for one of your solutions, use the reverse side of the page and indicate **clearly** which part of your work should be marked.)

Be aware that concise, well thought-out answers will be rewarded over long rambling ones. Also, unreadable answers will be given zero (0) so write legibly. In your answers, you may use without justification any facts given during the course, *as long as you state them clearly*. You must justify any other facts needed for your solutions.

General Hint: We were careful to leave ample space on the test paper to answer each question, so if you find yourself using much more room than what is available, you're probably missing something. Also, remember that hints are just hints: you are not required to follow them!
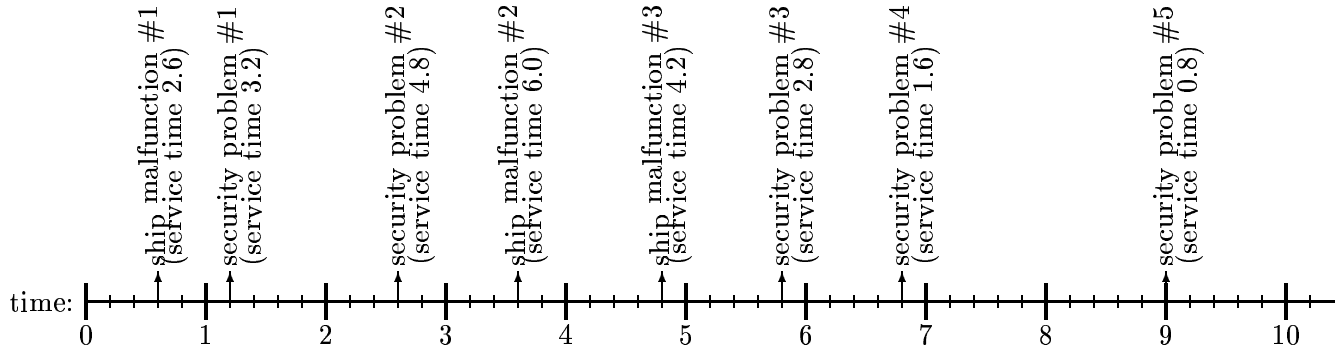
1: _____/25

2: _____/10

3: _____/15

TOTAL: _____/50

Good Luck!

Question 1.   [25 MARKS]

Consider the following event line depicting a sequence of "request for service" events for the starship simulation that you implemented in Assignment 2 (all times are indicated in minutes and each tick mark represents 0.2 minutes).

Part (a)   [5 MARKS]

In the diagram above, add the "end of service" events that would be generated during the simulation, assuming that we have 3 officers as in Assignment 2 and the command officer uses a "longest wait" strategy. (Draw the new events *below* the timeline, using a format similar to the one for the "request for service" events already there.)

Part (b)   [5 MARKS]

On what task is each officer working at time 10? (If not working, indicate that the officer is "idle".)

Engineer Officer: _____

Security Officer: _____

Command Officer: _____

Part (c)   [5 MARKS]

What are the contents of the two *waiting lists* at time 10?

Question 1.   (CONTINUED)

Part (d)   [5 MARKS]

Compute the average delay time for each waiting list, when the simulation reaches time 10. (You may leave your answer unsimplified but show your work; if you cannot finish your computation, at least explain what you were trying to do for part marks.)

Part (e)   [5 MARKS]

Compute the average utilization for each officer, when the simulation reaches time 10. (You may leave your answer unsimplified but show your work; if you cannot finish your computation, at least explain what you were trying to do for part marks.)

Question 2.   [10 MARKS]

Consider the following C++ code to define undirected graphs.

```
// Read the number of vertices and edges.
int num_vertices, num_edges;
cin >> num_vertices >> num_edges;

// Declare and initialize the graph's adjacencies.
bool **adj_matrix;
adj_matrix = new bool *[num_vertices];
for (int i = 0; i < num_vertices; i++) {
    adj_matrix[i] = new bool[num_vertices];
    for (int j = 0; j < num_vertices; j++)
        adj_matrix[i][j] = false;
} // end for

// Read the list of edges.
for (int e = 0; e < num_edges; e++) {
    int from, to;
    cin >> from >> to;
    adj_matrix[from][to] = true;
    adj_matrix[to][from] = true;
} // end for
```

Part (a)   [5 MARKS]

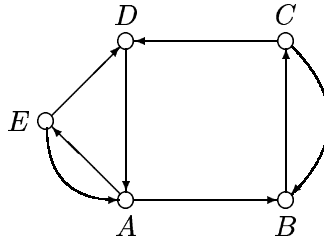Draw the graph that results from executing the code above on the following input.

```
6 6
0 3
1 2 1 4 1 5
3 0
5 2
```

Part (b)   [5 MARKS]

Write C++ code to free the memory that was dynamically allocated to store the graph. Assume that your code *immediately* follows the code above (in the same function, in the same file).

Question 3.   [15 MARKS]

Consider the following directed graph G.

Part (a)   [5 MARKS]

List the vertices of graph G in the order in which they would be visited by the breadth-first-search algorithm starting at vertex D. (Assume that adjacency lists are stored in alphabetical order.)

Part (b)   [5 MARKS]

List the edges of each simple cycle in graph G. (If G contains no simple cycle, simply state this fact.)

Part (c)   [3 MARKS]

Is graph G connected or not? *Briefly* justify your answer.

Part (d)   [2 MARKS]

Write your student number **legibly** at the top of every page of this test, *except page 1* (on page 1, write your student number only where we ask for it).