**CS2106 Introduction to Operating Systems**

**Lab 4**

**Contiguous Memory Allocation**

**Answer Book**

|  |  |  |
| --- | --- | --- |
| **Name:**  **Ling Guan Yu** | **Student ID:**  **A0308292A** | **Lab Group Number:**  **B20** |
| **Name:**  **NIL** | **Student ID:**  **NIL** | **Lab Group Number:**  **NIL** |

Question 3.1a (1 mark)

8 bytes (64 units to allocate [64 / 1] which corresponds to 64 bits and hence 8 bytes [64 / 8] as 1 byte is 8 bits)

Question 3.1b (1 mark)

1088 bytes (64 units to allocate [64 / 1] in the worst case which corresponds to 64 nodes and hence 1088 bytes [64 \* (4 + 4 + 4 + 4 + 1)])

Question 3.1c (1 mark)

Advantage: Lesser memory is used for the same number of units as each unit is represented by a single bit.

Disadvantage: Cannot contain more information than free/not free per unit as each unit is only represented by a single bit.

Question 3.2 (1 mark)

No, it would not make a difference as both (signed) char and unsigned char are 1 byte (8 bits) in size and the difference between the range of values they can represent does not affect anything.

Besides, choosing to use char (signed or unsigned) in the 1st place is probably just because it is the smallest primitive data type in C and hence using it results in space savings. In reality, using other data types that can represent the 2 states of allocated and freed will work too.

Question 3.3 (1 mark)

Yes, it needs to know how many bytes of memory need to be freed.

To get this information, I used the linked list library to create a static global linked list for keeping track of the length of each allocated block of memory using its “start” as the key. To get “start” in the myfree routine, I used the get\_index routine provided.

Question 4.1 (1 mark)

Question 4.2 (1 mark)

Question 4.3 (1 mark)

Question 4.4 (1 marks)

Question 5.1a (1 marks)

Question 5.1b (1 marks)

Question 5.2 (1 marks)

Question 5.3 (2 marks)

**TOTAL: \_\_\_\_\_\_\_\_\_\_\_ / 14**