1. **Given the following information:** <mark>**Please draw a picture to explain the process.**</mark>

Job List:	
Job Number	**Memory Requested**
Job A	690K
Job B	275K
Job C	760K

Memory Block List:	
Memory Block	**Memory Block Size**
Block 1	900K (low-order memory)
Block 2	910K
Block 3	300K (high-order memory)

   a. **Use the first-fit algorithm to indicate which memory blocks are allocated to each of the three arriving jobs.**

Job	記憶體需求	分配的記憶體區塊	區塊大小	工作大小	內部碎片
Job A	690K	Block 1	900K	690K	210K
Job B	275K	Block 3	300K	275K	25K
Job C	760K	Block 2	910K	760K	150K

可用空間合計: 900K + 300K + 910K = 2110K

用量合計: 690K + 275K + 760K = 1725K

總內部碎片: 210K + 25K + 150K = 385K

b. Use the best-fit algorithm to indicate which memory blocks are allocated to each of the three arriving jobs.

Job	記憶體需求	分配的記憶體區塊	區塊大小	工作大小	內部碎片
Job A	690K	Block 1	900K	690K	210K
Job B	275K	Block 3	300K	275K	25K
Job C	760K	Block 2	910K	760K	150K

總內部碎片 **: 210K + 25K + 150K = 385K**

用量合計: 690K + 275K + 760K = 1725K

可用空間合計: 900K + 300K + 910K = 2110K

2. **Next-fit is an allocation algorithm that starts by using the first-fit algorithm but keeps track of the partition that was last allocated, instead of restarting the search with Block 1, it starts searching from the most recently allocated block when a new job arrives. Using the following information:** <mark>**Please draw a picture to explain the process.**</mark>

Job List:	
Job Number	Memory Requested
Job A	590K
Job B	50K
Job C	275K
Job D	460K

Memory Block List:	
Memory Block	Memory Block Size
Block 1	100K (low-order memory)
Block 2	900K
Block 3	280K
Block 4	600K (high-order memory)

Indicate which memory blocks are allocated to each of the three arriving jobs, and explain in your own words what advantages the next-fit algorithm could offer.

Job	記憶體需求	分配的記憶體區塊	區塊大小	工作大小	內部碎片
Job A	590K	Block 2	900K	590K	310K
Job B	50K	Block 2	310K	50K	260k
Job C	275K	Block 3	280K	275K	5K
Job D	460K	Block 4	600K	460K	140K

可用空間合計: 900K + **280K + 600K** = 1780K.

用量合計: **590K + 50K + 275K** + 460 = 1375K.

總內部碎片 **: 310K + 260K + 5K = 140K**

減少搜尋時間，簡化實現，對於系統頻繁進行記憶體分配的場景，**Next-Fit** 可以提高系統的效率。