(Purpose: to mitigate fragmentation)

Placement Strategies - Best Fit. Worst Fit. First Fit Allocation

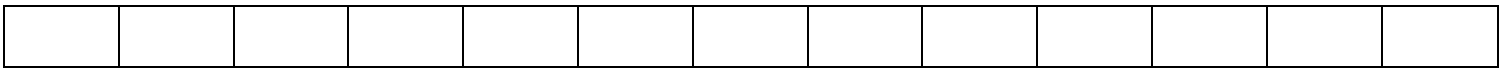
Suppose the heap is managed with a linked list. Each node in the list is either allocated or free. The list is sorted by address. When **`malloc()`** is called, the list is searched for a free segment that is big enough (depending on the allocation algorithm), that segment is divided into an allocated segment (at the beginning) and a free segment. When **`free()`** is called, the corresponding segment should merge with its neighboring segments, if they are also free. A process has a heap of 13KB, which is initially unallocated. During its execution, the process issues the following memory allocate/de-allocate calls (**`pA... pE`** are **`void*`** pointers). In all cases, break ties by choosing the earliest segment. Also, assume all algorithms allocate memory from the beginning of the free segment they choose.

```
pA = malloc(3KB)
pB = malloc(4KB)
pC = malloc(3KB)
free(pB)
pD = malloc(3KB)
free(pA)
pE = malloc(1KB)
```

For simplicity, assume the memory begins at address 0, and ignore the memory used by the linked list itself. Show the heap allocation after the above calls, using best-fit, worst-fit and first-fit algorithms respectively.

Best Fit:

0K 1K 2K 3K 4K 5K 6K 7K 8K 9K 10K 11K 12K



Starting address of pD= __3__ K and pE = __6__ K

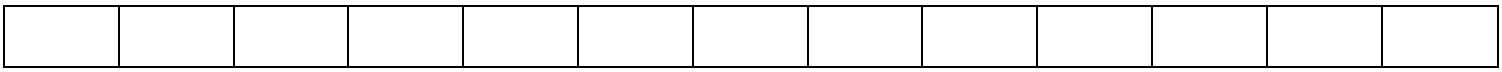
Worst Fit:

0K 1K 2K 3K 4K 5K 6K 7K 8K 9K 10K 11K 12K

Starting address of pD = __3__ K and pE = __0__ K

First Fit:

0K 1K 2K 3K 4K 5K 6K 7K 8K 9K 10K 11K 12K



Starting address of pD = __3__ K and pE = __0__ K

What is Fragmentation? What happens if heap memory is severely fragmented?

Best Fit outcome?

Worst Fit outcome?

First Fit outcome?P