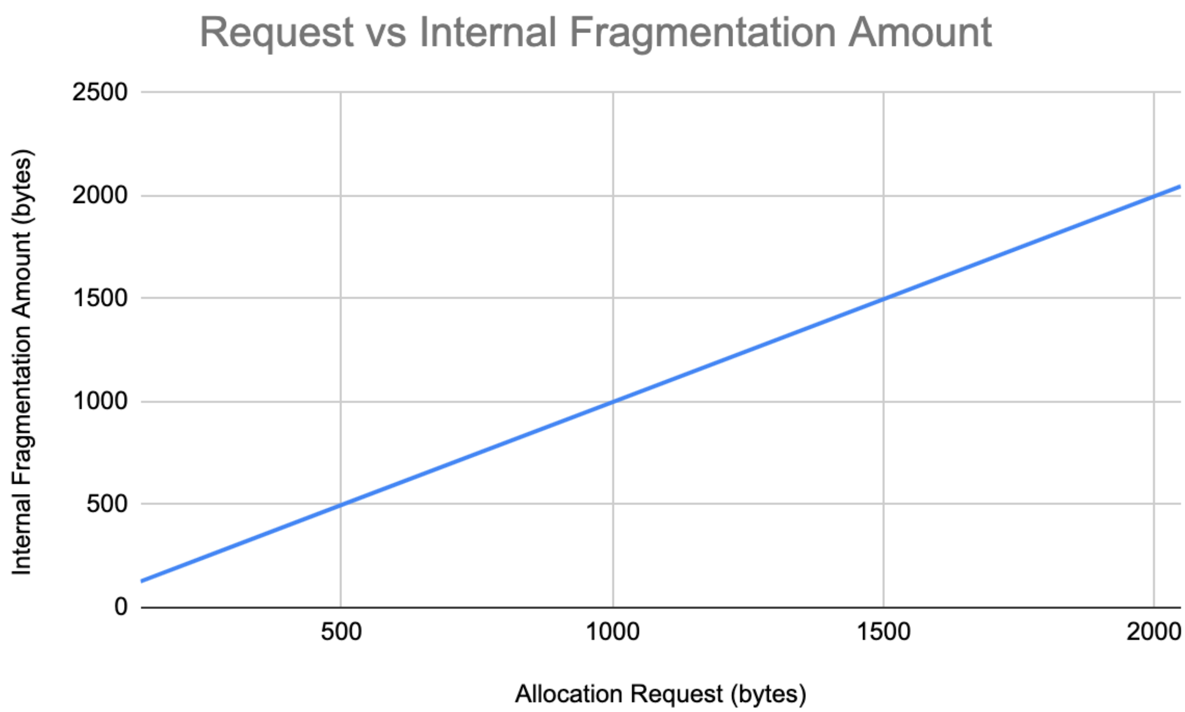
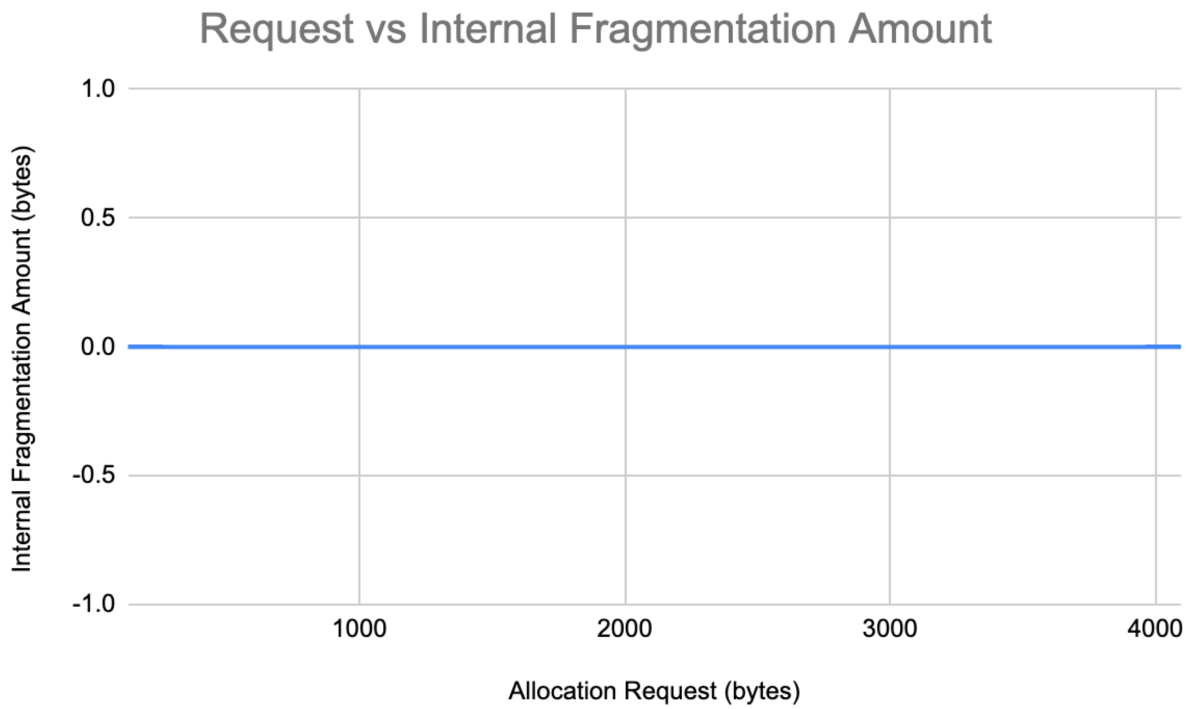
Osama Tanveer **21801147** **Section 3**
Adeem Adil Khatri **21801174** **Section 2**

Experiment 1



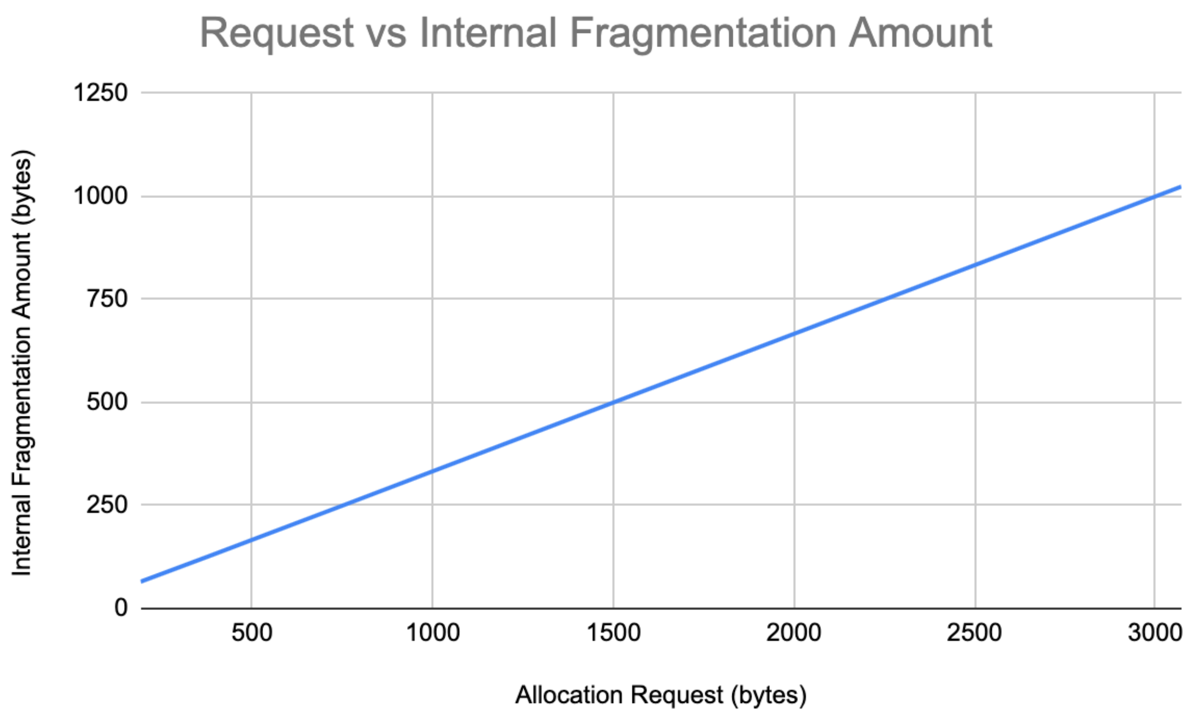
For the first experiment the allocation request was taken to match the worst-case scenario. This is the scenario in which the allocation request is $2^N + 1$ bytes, where N is an integer from 7 to 11 inclusive. The one byte overhead caused the buddy algorithm to allocate a block of size 2^N to just hold 1 byte, in addition to the previous 2^N bytes. This is the worst case of internal fragmentation. As it can be seen from the graph, as the allocation requests of this nature increase in size, the internal fragmentation increases almost as much as the request size. This is because for each request, its 2^N part is kept in a block and the additional 1 byte is kept in another block of size 2^N. The allocation is almost twice the request size.

Experiment 2



This experiment was performed using request sizes exactly the power of 2. The graph shows that there is no internal fragmentation as the size of the request is increased. This is because the buddy algorithm is able to find the perfectly sized block for this request. The amount of memory allocated is exactly the same as the request memory. This gives the best result in terms of no internal fragmentation.

Experiment 3



The third experiment performed involved using a request size that is exactly halfway between 2^N and 2^{N+1}. Even though an upward trend is observed, as was the case in experiment 1, it is important to realize that the gradient of the line is not as steep. This time the additional block that was allocated holds half its capacity, compared to experiment 1 where only 1 byte was being held. The graph is hence less steep and shows that internal fragmentation does not increase at the same rate as it did in experiment one.

Experiment 4

Below is an experiment conducted to demonstrate external fragmentation caused by the buddy allocation algorithm.

Let’s say the shared memory segment has a size of 32768 bytes and the inner 6 blocks of 4096 bytes are allocated to different processes. The only available memory left is the 4096 bytes at the start of the shared memory block and the 4096 at the end of the shared memory block. If a further 512 bytes and 2048 bytes are allocated from the first 4096 bytes, according to the buddy algorithm we are left with a space of 1536 bytes. This is because the memory from 0 to 511 address has been allocated for the 512 bytes request, and the memory from 2048 to 4095 has been allocated for the 2048 bytes request. Now suppose that in the last 4096 bytes of unallocated block, a request of 2048 bytes is granted. This leaves 2048 bytes of memory space unallocated. Even though the total allocated space 2048 + 1536 = 3584 bytes is unallocated, a request for this amount cannot be granted. 3584 bytes of external fragmentation has occurred. Even if the first 512 bytes are deallocated, a request of 4096 bytes cannot be granted. The sum of the available memory space is 4096 bytes, but a request for 4096 bytes cannot be fulfilled.

The allocation sequence of the algorithm is shown below.

```
void *p1 = sbmem_alloc(512);
p = sbmem_alloc(2048);

p = sbmem_alloc(4096);
p = sbmem_alloc(4096);
p = sbmem_alloc(4096);
p = sbmem_alloc(4096);
p = sbmem_alloc(4096);
p = sbmem_alloc(4096);

p = sbmem_alloc(2048);

sbmem_free(p1);
p = sbmem_alloc(4096);
```

```
Memory from 0 to 511 allocated.
Memory from 2048 to 4095 allocated.
Memory from 4096 to 8191 allocated.
Memory from 8192 to 12287 allocated.
Memory from 12288 to 16383 allocated.
Memory from 16384 to 20479 allocated.
Memory from 20480 to 24575 allocated.
Memory from 24576 to 28671 allocated.
Memory from 28672 to 30719 allocated.
Memory block 0 to 511 freed
No block available to allocate.
```

Conclusion

In conclusion, if the request size is likely to be a power of 2, the buddy algorithm is the perfect algorithm in terms of internal fragment. It has the potential to cause external fragmentation, but it is very less. In worst cases, buddy algorithm performs very poorly. However, in best cases, internal fragment is zero.