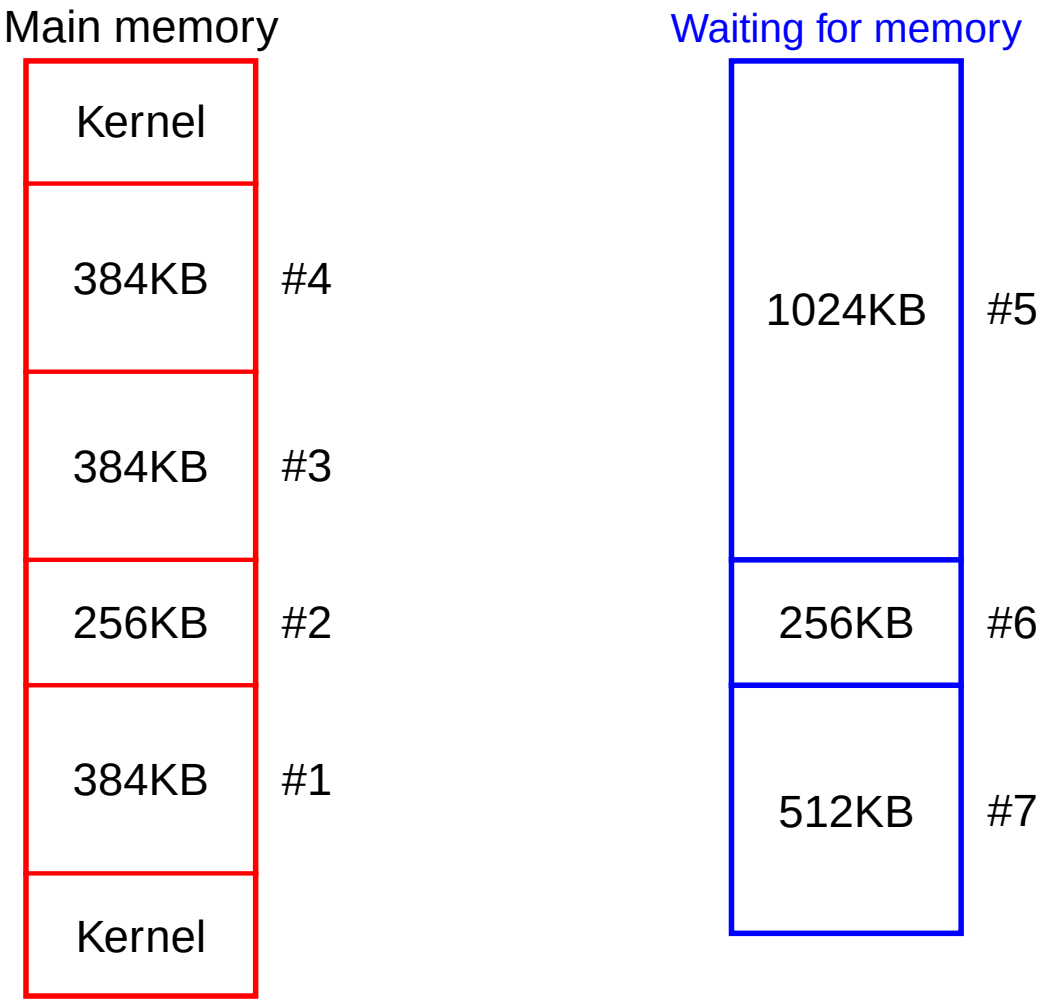
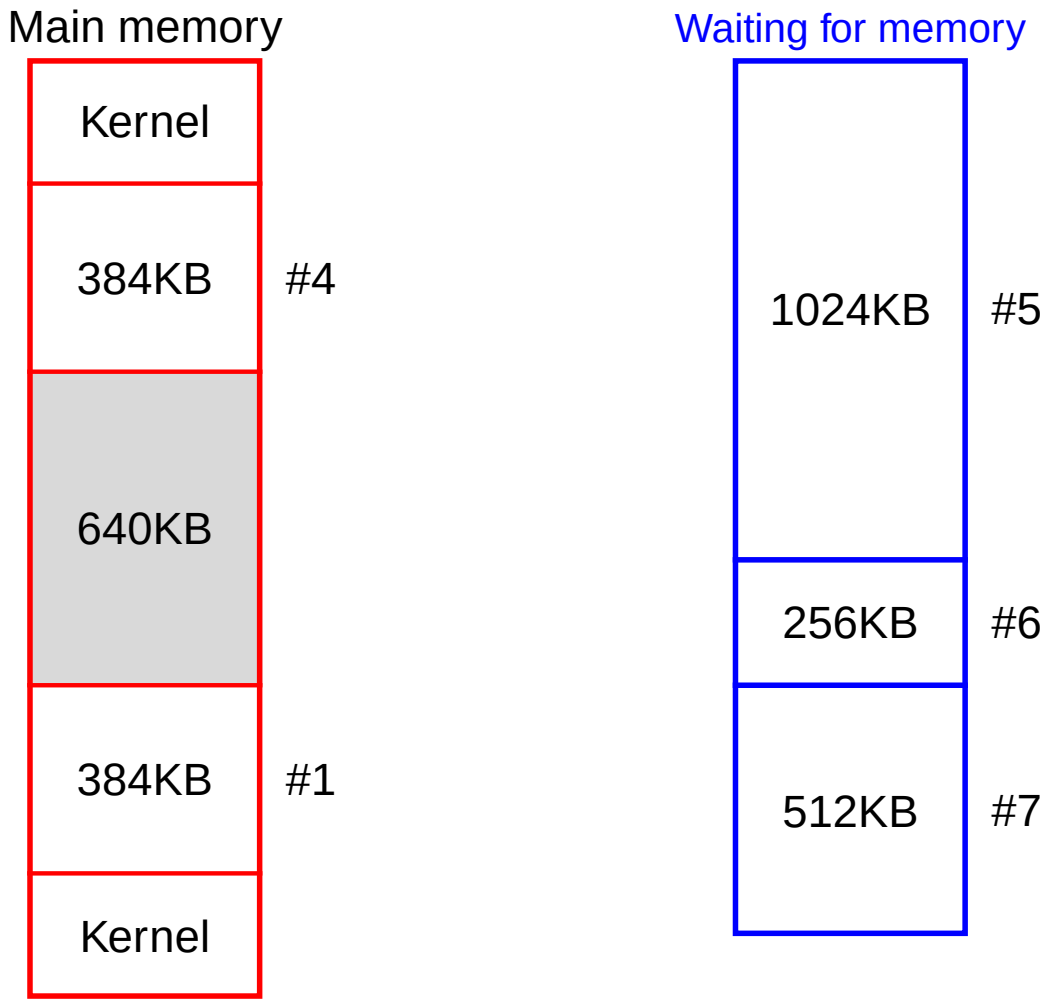
Describe the strengths and weaknesses of the following 3 methods of memory allocation in non–paged systems:

- Best fit: pick the free area that is the closest in size to the space needed.
- First fit: pick the free area that is the first found to fit.
- Worst fit: pick the free area that is the largest.

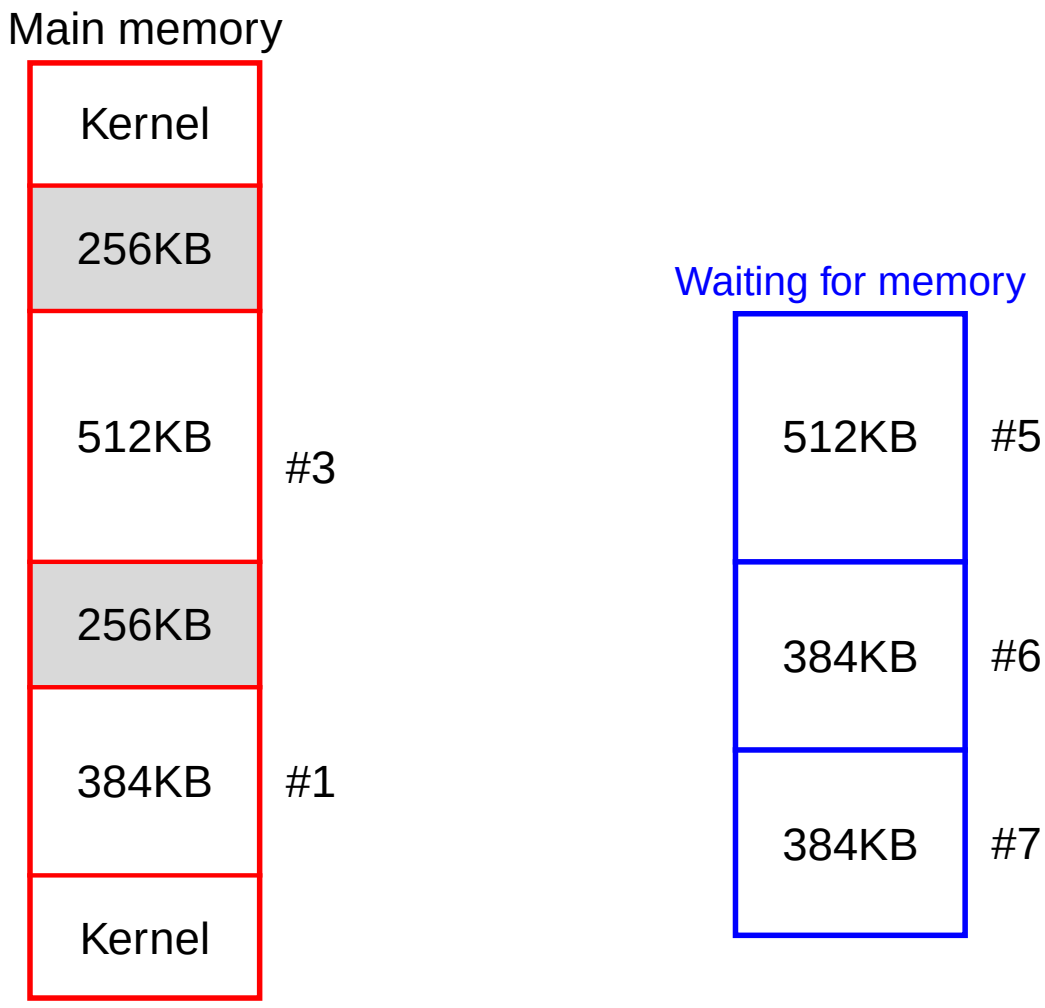
Main memory fragmentation



Main memory fragmentation



Main memory fragmentation



Reentrant code is a set of machine instructions that do not contain any direct memory references, i.e. all the memory references are done through a register, as in:

ST r1 , (r7) + r8

Any function not referencing global variables would be a good example of reentrant code, because all the memory references are relative to the stack but it is not if the machine instructions use direct addresses to point to other instructions (in loops, etc.).

- How to make code reentrant?
- What for?
- With 64–bit addressing, is it good to force programs to be reentrant?

Consider the instruction:

MOVE 0x415888, (r8)+0x24

which copies the value stored in 0x415888 to the memory location (r8)+0x24.

- How many page faults can it cause in a system with pure paging (no virtual memory)? How many disk transfers may be required?
- How many page faults can it cause in a virtual memory system? How many disk transfers may be required?
- How many segmentation faults can it cause? Assume a virtual memory system without paging.