Design:

All virtual memory codes are under system/mm. page.c and pageFault.S are used for handling paging mechanism and page fault. service.c is used to handle virtual memory management such as vmcreat, vgetmem, etc.

Bonus Problem:

This is implemented as function pGClk().

Here, we implemented the global clock page replacement.

At first, we test on the micro benchmark, then create one process and limit the number of physical frames for paging to be 4. But page table allocation will not be limited.

Code Snippet:

```
ptr = vgetmem(20480);

*ptr = 11;
*(ptr + PAGE_SIZE) = 22;
*(ptr + PAGE_SIZE * 2) = 33;
*(ptr + PAGE_SIZE * 3) = 44;
*(ptr + PAGE_SIZE * 4) = 55;
kprintf("0x%x %d\n", ptr, *ptr);
kprintf("0x%x %d\n", ptr + PAGE_SIZE, *(ptr + PAGE_SIZE));
kprintf("0x%x %d\n", ptr + PAGE_SIZE * 2, *(ptr + PAGE_SIZE * 2));
kprintf("0x%x %d\n", ptr + PAGE_SIZE * 3, *(ptr + PAGE_SIZE * 3));
kprintf("0x%x %d\n", ptr + PAGE_SIZE * 4, *(ptr + PAGE_SIZE * 4));
*(ptr + PAGE_SIZE * 2) = 33;
*(ptr + PAGE_SIZE * 3) = 44;
*(ptr + PAGE_SIZE * 4) = 55;
kprintf("0x%x %d\n", ptr + PAGE_SIZE, *(ptr + PAGE_SIZE));
kprintf("0x%x %d\n", ptr, *ptr);
*ptr = 11;
*(ptr + PAGE_SIZE * 4) = 55;
*(ptr + PAGE_SIZE) = 22;
```

We use vgetmem to get 20480 bytes(i.e. 5 pages from 4096 to 4100) from virtual memory at first. Then we write different values to the first byte of each page one by one. There should be six pages being used (5 + pointer used by virtual memory allocator). Then we can find out that page fault sequence without calling global clock is vpn

4096, 4101, 4097, 4098. We have 4101 here is because vgetmem will allocate memory from vpn 4096~4100, then memory list will link to page 4101 and its starting 8 bytes we be written.
Then, when we write to 4099, we need to swap one page out. By the algorithm of global clock, we can see that virtual page 4096 will be swapped out and replaced by 4099. Then the pointer of global clock stop at 4101. Then when we write to 4100, page 4101 will be swapped out.

At this time, all pages'(4097, 4098, 4099) dirty and access bits are cleared by global clock except page 4100. Thus, we write to vpn 4098, 4099 and read from 4097. Then we can incur another page fault by accessing vpn 4096. Then, we can see page 4097 will be swapped out because it has only access flag set.

Finally, we write to page 4100 again and make it dirty. When we access vpn 4097 again, we can see that page 4098 will be swapped out because global clock start scanning from 4098's physical frame and changes (dirty, access) bits of vpn 4098, 4099 to zero. Moreover, (dirty, access) bits 4096 and 4100 are changed from (1, 1) to (0, 1). Then because global clock start from vpn 4098's physical frame, it will be swapped out to be replaced by 4097.

For big program testing, we create a process which allocate 12582912(i.e. 3072 pages) bytes from virtual memory. This means we will need more than 3072 physical frames because of memory for page tables and for other system processes. Then we write from page 0 to page 3072 and read them from 0 to 3072 again. From the output, we can see the difference between global clock and FIFO is that FIFO will incur page fault for each page R/W. Thus, FIFO's page fult number is 6145, and global clock's page fault number is 6104.