**CSCE 613: Project 2 Design**

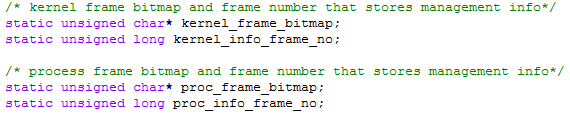
**Student: Caio Duarte Diniz Monteiro**

## Date: 02/05/16

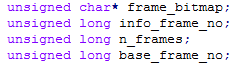
# FramePool class variables

The frame pool is responsible for managing the physical memory frame allocation for both processes and the kernel. While we have different frame pool instances for the kernel and for the process, the *release\_frame* method is static, thus, the *FramePool* class needs to keep track of all the existent pools.

The *FramePool* class have 4 static variables, two of them to hold the addresses of the kernel and process frames bitmaps and other two that holds the info frame numbers for the kernel and process pools.



Furthermore, the *FramePool* class has more 4 variables that holds all the information needed by a particular instance of this class.



The *frame\_bitmap* holds the address of the bitmap for the current instance, *info\_frame\_no* holds the frame number where the bitmap is stored, *n\_frames* tells how many frames this pool is responsible to manage, and the *base\_frame\_no* tells the starting frame of the current frame pool.

# FramePool initialization



# FramePool class functions

In order to manage the physical memory allocation, the pool must be able to perform three actions: (i) allocate a new frame; (ii) release an allocated frame; (iii) protect specific parts of the memory.

## get\_frame()



## release\_frame



## mark\_inaccessible

