CSCE 410 HW2 Design Document
Taylor Williamson
125007948
2/15/2020

**Bitmap Implementation:**

To implement control functions over the frame pools, we need to establish a framework in which the status of all elements of the frame pool will be stored and maintained. The basis of this framework is a bitmap made up of 8-bit characters. For this homework, it is required to determine the status of the frame (free or allocated), and also indicate whether that frame is the first frame in a contiguous sequence of frames, called head-of-sequence. Therefore, two bits per frame are needed to store the status for every frame in the pool. Furthermore, it can be determined that each character object can maintain the status of four frames each.

All bits in the bitmap will be initialized to 1, to follow the convention implemented in the simple frame pool example. I have made the decision that the first bit of each bit pair will represent the free bit, and the second will be the head-of-sequence indicator.

**Function Implementations:**

get_frames(n_frames):

- Checks if there are enough frames to allocate
- Skips to first set of frames that are not all allocated
- Checks if there are enough frames succeeding it to properly allocate the requested number of frames
- It checks the rest of the bitmap until it can find a window of frames where n_frames can fit.
- If one does not exist, message is sent to the console notifying the user that it cannot be allocated
    - Or allocated frames can be shifted towards the front to free up enough space at the end to allocate the requested frames.
- Helper functions:
    - check_sequence(first_frame, n_frames)
        - Checks for a window of n free frames

release_frames(first_frame_no):

- Check if the frame number is valid and in range
- Check if the frame is allocated and head-of-sequence
- Update all frames from the first frame up to the next head-of-sequence frame or unallocated frame in the pool to indicate they are unallocated