Alexey Shablygin
CS377 Lab 3 Part 1
04/23/16
README

Part 1 is written in python (2.7) and works as follows.

First the program reads pre-defined input file, constructing an array of arrays. Then we pop first two elements - these are n (offset) and m (page number). Everything else left in our data structure are virtual addresses.

Since we know what bits of the virtual addresses correspond to offset and page number, the quickest way to extract those bits is to create two custom masks - one for extracting n bits and one for extracting m bits. Since n bits are least significant bits, we create a mask with m leading 0 and append n 1. Then by using built-in python function *int()* we convert our binary string into an integer and return it as our n mask. We do the same process for m_mask, except we swap n and m in place (also make a note that we need to shift right by n later on. After we have our masks, we start reading virtual addresses from the input file. In order to get the offset (n) we take our virtual address and perform bitwise AND operation with the n_mask. To get the page number, we take the same virtual address, perform the bitwise AND operation with the m_mask and bitshift right by n bits.

Currently the program is written to work with only one hardcoded input file, but it could work with an array of input files by putting the *main()* routine into a proper for loop.
