Homework: mmap()

This assignment will make you more familiar with how to manage virtual memory in user programs using the Unix system call interface. You can do this assignment on any operating system that supports the Unix API (a Linux Athena machine, your laptop with Linux or MacOS, etc.). Submit your solution to the submission web site as a text file with the name "hwN.c", where N is the homework number as listed on the schedule.

Download the mmap homework assignment and look it over. The program maintains a very large table of square root values in virtual memory. However, the table is too large to fit in physical RAM. Instead, the square root values should be computed on demand in response to page faults that occur in the table's address range. Your job is to implement the demand faulting mechanism using a signal handler and UNIX memory mapping system calls. To stay within the physical RAM limit, we suggest using the simple strategy of unmapping the last page whenever a new page is faulted in.

To compile `mmap.c`, you need a C compiler, such as gcc. On Athena, you can type:

```
$ add gnu
```

Once you have gcc, you can compile mmap.c as follows:

```
$ gcc mmap.c -lm -o mmap
```

Which produces a `mmap` file, which you can run:

```
$ ./mmap
page_size is 4096
Validating square root table contents...
oops got SIGSEGV at 0x7f6bf7fd7f18
```

When the process accesses the square root table, the mapping does not exist and the kernel passes control to the signal handler code in `handle_sigsegv()`. Modify the code in `handle_sigsegv()` to map in a page at the faulting address, unmap a previous page to stay within the physical memory limit, and initialize the new page with the correct square root values. Use the function `calculate_sqrts()` to compute the values. The program includes test logic that verifies if the contents of the square root table are correct. When you have completed your task successfully, the process will print “All tests passed!”.

You may find that the man pages for mmap() and munmap() are helpful references.

```
$ man mmap
$ man munmap
```

Don't forget to submit your solution to the submission web site (using the "hwN.c" naming convention as outlined above).