PSUEDO CODE

```
        //user inputs
        System.out.print: page reference pattern length?
                patternlength=scanner
        System.out.print: unique pages?
                Number of pages =scanner
        System.out.print: slots?
                Number of slots=scanner
System print (“Ref str:” Reference string ) with format
Random rand = random()
        Reference string = random(char)
Hash all algorithms to a map;
        -each algorithm will be a new empty linked list, with they key being the string name

-Iterate through algorithms with key:
        -set up a cache linked list to store our letters per algorithm
-for(I ; I < number of slots; ++)
        -FIFO algorithm () {
        “+” if in page is in cache
                .removeFirst()
                .add() to
        Print char if miss or fault
        }
        LRU () {
        “+” if in page is in cache
                .removeLast()?
                .add()
        Print char if miss or fault
```

}

MIN () {

“+” if in page is in cache

Iterate through index of character in string to find lowest index with for loop

Set lowest to newest index and .add() to cache

Print char if miss or fault

}

RAND() {

“+” if in page is in cache

Random rand = random();

Index = random(of size of cache)

Replace index

Print char if miss or fault

}

Main static {

Hash keys to print each algorithm: strings

Print cache hit rates;

Print(worst : algorithm

Best: algorithm)

}