CS 525: Advance Database Organization

PROGRAMMING ASSIGNMENT 2: BUFFER MANAGER

CS525-F24-G17

Group #17 Members:

- Adarsh Chidirala - A20561069 - 25% Contributed
- Venkata Naga Lakshmi Sai Snigdha Sri Jata - A20560684 - 25% Contributed
- Sharan Rama Prakash Shenoy - A20560683 - 25% Contributed
- Ajay Kumar Choudary Koneti - A20563634 - 25% Contributed

README- Buffer Manager

This README provides a comprehensive guide to understanding and executing the Buffer Manager code, covering everything from buffer pool initialization to error handling. This offers a comprehensive guide to the Buffer Manager code, segmented into three key sections: Pool Handling Interface, Access Pages Interface, and Replacement Strategies along with Statistics Interface. Each of these sections elucidates the role and functionality of the associated functions and methods.

Additional Extensions:

We have implemented additional page replacement strategies **LRU_k** & **CLOCK** with required test cases(test case 2).

The Procedure to Run the Code:

Step 1: Navigate to the new branch created "**assign2**" branch in Git-Hub and download the necessary .zip file.

Step 2: Run "***make clean***" to remove any previously compiled files.

Step 3: Use the "***make***" command to compile the program by running the **Makefile**.

Step 4: Execute the test case 1 by running "***./g17_test1***".

Step 5: Execute the test case 2 by running "***./g17_test2***".

Step 6: Once again, use "***make clea*n**" to remove generated executable files, and repeat steps 3 and 5 to rerun the assignment.

Features To Do

1. **Buffer Pool Initialization:** Initialize a new buffer pool with a specific page file, page size, and replacement strategy.
2. **Buffer Pool Disposal:** Free up all resources associated with a buffer pool.
3. **Page Management:** Manage the reading, writing, and pinning of pages in the buffer pool.
    a. Pin Page: Load a page into the buffer pool.
    b. Unpin Page: Release a page from the buffer pool.
    c. Force Page: Immediately write a page back to disk.

    d. Mark Dirty: Mark a page as modified.
4. **Replacement Strategies:** Implement various page replacement strategies such as FIFO, LRU, LRU_k and CLOCK.
    a. "Statistics:" Provide statistics about the buffer pool, including:
    b. Get Frame Contents: Get an array of PageNumbers holding the contents of the page frames.
    c. Get Dirty Flags: Get an array of bools representing whether the page frames are dirty.
    d. Get Fix Counts: Get an array of ints showing the fix count of page frames.
    e. Get Number of Reads: Get the number of pages that have been read from disk.
    f. Get Number of Writes: Get the number of pages that have been written to disk.
5. **Error Handling:** Comprehensive error handling for invalid buffer pool operations, out-of-range pages, and failed reads/writes.

BUFFER MANAGER

PART 1: BUFFER POOL INITIALIZATION

This section explains the functions used to initialize a new buffer pool with a specific page file, page size, and replacement strategy.

void initBufferPool(BM_BufferPool *const bm, const char *const pageFileName, const int numPages, ReplacementStrategy strategy, void *stratData)

Initializes a new buffer pool with the given parameters. It allocates memory for the buffer pool and initializes the necessary data structures. The replacement strategy and strategy data are also set.

void shutdownBufferPool(BM_BufferPool *const bm)

Shuts down the buffer pool, freeing up all resources associated with it. It writes dirty pages back to disk if necessary and deallocates memory.

RC forceFlushPool(BM_BufferPool *const bm)

Forces all dirty pages in the buffer pool to be written back to disk. It iterates through the buffer pool and writes each dirty page to disk.

PART 2: ACCESS PAGES INTERFACE

This section describes the functions used to manage the reading, writing, and pinning of pages in the buffer pool.

RC markDirty(BM_BufferPool *const bm, BM_PageHandle *const page)

Marks a page in the buffer pool as dirty. It sets the dirty flag of the page and updates the fix count.

RC unpinPage(BM_BufferPool *const bm, BM_PageHandle *const page)

Unpins a page in the buffer pool. It decreases the fix count of the page.

RC forcePage(BM_BufferPool *const bm, BM_PageHandle *const page)

Forces a page in the buffer pool to be written back to disk. It writes the page to disk if it is dirty.

RC pinPage(BM_BufferPool *const bm, BM_PageHandle *const page, const PageNumber pageNum)

Pins a page in the buffer pool. It loads the page into the buffer pool if it is not already present and increases the fix count.

PART 3: REPLACEMENT STRATEGIES

This section explains the various page replacement strategies implemented in the buffer manager.

RC FIFO(BM_BufferPool *const bm, BM_PageHandle *const page)

Implements the First-In-First-Out (FIFO) page replacement strategy. It selects the oldest unpinned page in the buffer pool for replacement.

RC LRU(BM_BufferPool *const bm, BM_PageHandle *const page)

Implements the Least Recently Used (LRU) page replacement strategy. It selects the least recently used unpinned page in the buffer pool for replacement.

RC LRU_k(BM_BufferPool *const bm, BM_PageHandle *const page)

Implements the LRU_k page replacement strategy. It selects the least recently used unpinned page in the buffer pool for replacement, considering the last k page requests.

RC CLOCK(BM_BufferPool *const bm, BM_PageHandle *const page)

Implements the CLOCK page replacement strategy. It selects the next unpinned page in the buffer pool for replacement, using a clock hand to keep track of the pages.

PART 4: STATISTICS INTERFACE

This section describes the functions used to provide statistics about the buffer pool.

PageNumber *getFrameContents(BM_BufferPool *const bm)

Returns an array of PageNumbers holding the contents of the page frames in the buffer pool.

bool *getDirtyFlags(BM_BufferPool *const bm)

Returns an array of bools representing whether the page frames in the buffer pool are dirty.

int *getFixCounts(BM_BufferPool *const bm)

Returns an array of ints showing the fix count of the page frames in the buffer pool.

int getNumReadIO(BM_BufferPool *const bm)

Returns the number of pages that have been read from disk.

int getNumWriteIO(BM_BufferPool *const bm)

Returns the number of pages that have been written to disk.

PART 5: ERROR HANDLING

This section explains the comprehensive error handling implemented in the buffer manager. It handles invalid buffer pool operations, out-of-range pages, and failed reads/writes.

MemoryLeak Check:

There are no memory leaks in the code. Below are commands used to check.

- **MAC OS:**

  - ***`leaks -atExit -- ./g17_test1 | grep LEAK:`***
  - ***`leaks -atExit -- ./g17_test2 | grep LEAK:`***

```
adarshchidirala@dhcp222 ADO_Assign2_Group17 % leaks -atExit -- ./g17_test1 | grep LEAK:
g17_test1(27344) MallocStackLogging: could not tag MSL-related memory as no_footprint, so those pages will be included in process footprint - (null)
g17_test1(27344) MallocStackLogging: recording malloc (and VM allocation) stacks using lite mode
Process 27344 is not debuggable. Due to security restrictions, leaks can only show or save contents of readonly memory of restricted processes.

adarshchidirala@dhcp222 ADO_Assign2_Group17 % leaks -atExit -- ./g17_test2 | grep LEAK:
g17_test2(27361) MallocStackLogging: could not tag MSL-related memory as no_footprint, so those pages will be included in process footprint - (null)
g17_test2(27361) MallocStackLogging: recording malloc (and VM allocation) stacks using lite mode
Process 27361 is not debuggable. Due to security restrictions, leaks can only show or save contents of readonly memory of restricted processes.
```

OUTPUT SCREENSHOTS:

After compiling the code with "make" then type "./g17_test1" to run the test_assign2_1.c file.

1. Testing Assign2_1 (test_assign2_1.c):

1. Creating and checking 10000 dummy pages:

```
 <Page-9986>: reading back dummy page content
[test_assign2_1.c-Creating and Reading Back Dummy Pages-L111-12:05:21] OK: expected <Page-9987> and was
 <Page-9987>: reading back dummy page content
[test_assign2_1.c-Creating and Reading Back Dummy Pages-L111-12:05:21] OK: expected <Page-9988> and was
 <Page-9988>: reading back dummy page content
[test_assign2_1.c-Creating and Reading Back Dummy Pages-L111-12:05:21] OK: expected <Page-9989> and was
 <Page-9989>: reading back dummy page content
[test_assign2_1.c-Creating and Reading Back Dummy Pages-L111-12:05:21] OK: expected <Page-9990> and was
 <Page-9990>: reading back dummy page content
[test_assign2_1.c-Creating and Reading Back Dummy Pages-L111-12:05:21] OK: expected <Page-9991> and was
 <Page-9991>: reading back dummy page content
[test_assign2_1.c-Creating and Reading Back Dummy Pages-L111-12:05:21] OK: expected <Page-9992> and was
 <Page-9992>: reading back dummy page content
[test_assign2_1.c-Creating and Reading Back Dummy Pages-L111-12:05:21] OK: expected <Page-9993> and was
 <Page-9993>: reading back dummy page content
[test_assign2_1.c-Creating and Reading Back Dummy Pages-L111-12:05:21] OK: expected <Page-9994> and was
 <Page-9994>: reading back dummy page content
[test_assign2_1.c-Creating and Reading Back Dummy Pages-L111-12:05:21] OK: expected <Page-9995> and was
 <Page-9995>: reading back dummy page content
[test_assign2_1.c-Creating and Reading Back Dummy Pages-L111-12:05:21] OK: expected <Page-9996> and was
 <Page-9996>: reading back dummy page content
[test_assign2_1.c-Creating and Reading Back Dummy Pages-L111-12:05:21] OK: expected <Page-9997> and was
 <Page-9997>: reading back dummy page content
[test_assign2_1.c-Creating and Reading Back Dummy Pages-L111-12:05:21] OK: expected <Page-9998> and was
 <Page-9998>: reading back dummy page content
[test_assign2_1.c-Creating and Reading Back Dummy Pages-L111-12:05:21] OK: expected <Page-9999> and was
 <Page-9999>: reading back dummy page content
[test_assign2_1.c-Creating and Reading Back Dummy Pages-L72-12:05:21] OK: finished test

[test_assign2_1.c-Reading a page-L148-12:05:21] OK: finished test
```

2. Testing FIFO Replacement Strategy:

```
[test_assign2_1.c-Testing FIFO page replacement-L188-12:05:21] OK: expected <[0 0],[-1 0],[-1 0]> and w
as <[0 0],[-1 0],[-1 0]>: check pool content
[test_assign2_1.c-Testing FIFO page replacement-L188-12:05:21] OK: expected <[0 0],[1 0],[-1 0]> and wa
s <[0 0],[1 0],[-1 0]>: check pool content
[test_assign2_1.c-Testing FIFO page replacement-L188-12:05:21] OK: expected <[0 0],[1 0],[2 0]> and was
 <[0 0],[1 0],[2 0]>: check pool content
[test_assign2_1.c-Testing FIFO page replacement-L188-12:05:21] OK: expected <[3 0],[1 0],[2 0]> and was
 <[3 0],[1 0],[2 0]>: check pool content
[test_assign2_1.c-Testing FIFO page replacement-L188-12:05:21] OK: expected <[3 0],[4 0],[2 0]> and was
 <[3 0],[4 0],[2 0]>: check pool content
[test_assign2_1.c-Testing FIFO page replacement-L194-12:05:21] OK: expected <[3 0],[4 1],[2 0]> and was
 <[3 0],[4 1],[2 0]>: pool content after pin page
[test_assign2_1.c-Testing FIFO page replacement-L202-12:05:21] OK: expected <[3 0],[4 1],[5x0]> and was
 <[3 0],[4 1],[5x0]>: check pool content
[test_assign2_1.c-Testing FIFO page replacement-L202-12:05:21] OK: expected <[6x0],[4 1],[5x0]> and was
 <[6x0],[4 1],[5x0]>: check pool content
[test_assign2_1.c-Testing FIFO page replacement-L202-12:05:21] OK: expected <[6x0],[4 1],[0x0]> and was
 <[6x0],[4 1],[0x0]>: check pool content
[test_assign2_1.c-Testing FIFO page replacement-L209-12:05:21] OK: expected <[6x0],[4 0],[0x0]> and was
 <[6x0],[4 0],[0x0]>: unpin last page
[test_assign2_1.c-Testing FIFO page replacement-L213-12:05:21] OK: expected <[6 0],[4 0],[0 0]> and was
 <[6 0],[4 0],[0 0]>: pool content after flush
[test_assign2_1.c-Testing FIFO page replacement-L216-12:05:21] OK: expected <3> and was <3>: check numb
er of write I/Os
[test_assign2_1.c-Testing FIFO page replacement-L217-12:05:21] OK: expected <8> and was <8>: check numb
er of read I/Os
[test_assign2_1.c-Testing FIFO page replacement-L224-12:05:21] OK: finished test
```

3. Testing LRU Replacement Strategy:

```
[test_assign2_1.c-Testing LRU page replacement-L270-12:05:21] OK: expected <[0 0],[-1 0],[-1 0],[-1 0],
[-1 0]> and was <[0 0],[-1 0],[-1 0],[-1 0],[-1 0]>: check pool content reading in pages
[test_assign2_1.c-Testing LRU page replacement-L270-12:05:21] OK: expected <[0 0],[1 0],[-1 0],[-1 0],[
-1 0]> and was <[0 0],[1 0],[-1 0],[-1 0],[-1 0]>: check pool content reading in pages
[test_assign2_1.c-Testing LRU page replacement-L270-12:05:21] OK: expected <[0 0],[1 0],[2 0],[-1 0],[-
1 0]> and was <[0 0],[1 0],[2 0],[-1 0],[-1 0]>: check pool content reading in pages
[test_assign2_1.c-Testing LRU page replacement-L270-12:05:21] OK: expected <[0 0],[1 0],[2 0],[3 0],[-1
 0]> and was <[0 0],[1 0],[2 0],[3 0],[-1 0]>: check pool content reading in pages
[test_assign2_1.c-Testing LRU page replacement-L270-12:05:21] OK: expected <[0 0],[1 0],[2 0],[3 0],[4
0]> and was <[0 0],[1 0],[2 0],[3 0],[4 0]>: check pool content reading in pages
[test_assign2_1.c-Testing LRU page replacement-L279-12:05:21] OK: expected <[0 0],[1 0],[2 0],[3 0],[4
0]> and was <[0 0],[1 0],[2 0],[3 0],[4 0]>: check pool content using pages
[test_assign2_1.c-Testing LRU page replacement-L279-12:05:21] OK: expected <[0 0],[1 0],[2 0],[3 0],[4
0]> and was <[0 0],[1 0],[2 0],[3 0],[4 0]>: check pool content using pages
[test_assign2_1.c-Testing LRU page replacement-L279-12:05:21] OK: expected <[0 0],[1 0],[2 0],[3 0],[4
0]> and was <[0 0],[1 0],[2 0],[3 0],[4 0]>: check pool content using pages
[test_assign2_1.c-Testing LRU page replacement-L279-12:05:21] OK: expected <[0 0],[1 0],[2 0],[3 0],[4
0]> and was <[0 0],[1 0],[2 0],[3 0],[4 0]>: check pool content using pages
[test_assign2_1.c-Testing LRU page replacement-L279-12:05:21] OK: expected <[0 0],[1 0],[2 0],[3 0],[4
0]> and was <[0 0],[1 0],[2 0],[3 0],[4 0]>: check pool content using pages
[test_assign2_1.c-Testing LRU page replacement-L288-12:05:21] OK: expected <[0 0],[1 0],[2 0],[5 0],[4
0]> and was <[0 0],[1 0],[2 0],[5 0],[4 0]>: check pool content using pages
[test_assign2_1.c-Testing LRU page replacement-L288-12:05:21] OK: expected <[0 0],[1 0],[2 0],[5 0],[6
0]> and was <[0 0],[1 0],[2 0],[5 0],[6 0]>: check pool content using pages
[test_assign2_1.c-Testing LRU page replacement-L288-12:05:21] OK: expected <[7 0],[1 0],[2 0],[5 0],[6
0]> and was <[7 0],[1 0],[2 0],[5 0],[6 0]>: check pool content using pages
[test_assign2_1.c-Testing LRU page replacement-L288-12:05:21] OK: expected <[7 0],[1 0],[8 0],[5 0],[6
0]> and was <[7 0],[1 0],[8 0],[5 0],[6 0]>: check pool content using pages
[test_assign2_1.c-Testing LRU page replacement-L288-12:05:21] OK: expected <[7 0],[9 0],[8 0],[5 0],[6
0]> and was <[7 0],[9 0],[8 0],[5 0],[6 0]>: check pool content using pages
[test_assign2_1.c-Testing LRU page replacement-L293-12:05:21] OK: expected <0> and was <0>: check numbe
r of write I/Os
[test_assign2_1.c-Testing LRU page replacement-L294-12:05:21] OK: expected <10> and was <10>: check num
ber of read I/Os
[test_assign2_1.c-Testing LRU page replacement-L301-12:05:21] OK: finished test
```

2. Testing Assign2_2 (test_assign2_2.c):

Then run "./g17_test2" for test_assign2_2.c file.

1. Testing LRU_k Replacement Strategy:

```
adarshchidirala@dhcp222 ADO_Assign2_Group17 % ./g17_test2
[test_assign2_2.c-Testing LRU_K page replacement-L118-12:05:21] OK: expected <[0 0],[-1 0],[-1 0],[-1 0
],[-1 0]> and was <[0 0],[-1 0],[-1 0],[-1 0],[-1 0]>: check pool content reading in pages
[test_assign2_2.c-Testing LRU_K page replacement-L118-12:05:21] OK: expected <[0 0],[1 0],[-1 0],[-1 0]
,[-1 0]> and was <[0 0],[1 0],[-1 0],[-1 0],[-1 0]>: check pool content reading in pages
[test_assign2_2.c-Testing LRU_K page replacement-L118-12:05:21] OK: expected <[0 0],[1 0],[2 0],[-1 0],
[-1 0]> and was <[0 0],[1 0],[2 0],[-1 0],[-1 0]>: check pool content reading in pages
[test_assign2_2.c-Testing LRU_K page replacement-L118-12:05:21] OK: expected <[0 0],[1 0],[2 0],[3 0],[
-1 0]> and was <[0 0],[1 0],[2 0],[3 0],[-1 0]>: check pool content reading in pages
[test_assign2_2.c-Testing LRU_K page replacement-L118-12:05:21] OK: expected <[0 0],[1 0],[2 0],[3 0],[
4 0]> and was <[0 0],[1 0],[2 0],[3 0],[4 0]>: check pool content reading in pages
[test_assign2_2.c-Testing LRU_K page replacement-L126-12:05:21] OK: expected <[0 0],[1 0],[2 0],[3 0],[
4 0]> and was <[0 0],[1 0],[2 0],[3 0],[4 0]>: check pool content using pages
[test_assign2_2.c-Testing LRU_K page replacement-L126-12:05:21] OK: expected <[0 0],[1 0],[2 0],[3 0],[
4 0]> and was <[0 0],[1 0],[2 0],[3 0],[4 0]>: check pool content using pages
[test_assign2_2.c-Testing LRU_K page replacement-L126-12:05:21] OK: expected <[0 0],[1 0],[2 0],[3 0],[
4 0]> and was <[0 0],[1 0],[2 0],[3 0],[4 0]>: check pool content using pages
[test_assign2_2.c-Testing LRU_K page replacement-L126-12:05:21] OK: expected <[0 0],[1 0],[2 0],[3 0],[
4 0]> and was <[0 0],[1 0],[2 0],[3 0],[4 0]>: check pool content using pages
[test_assign2_2.c-Testing LRU_K page replacement-L126-12:05:21] OK: expected <[0 0],[1 0],[2 0],[3 0],[
4 0]> and was <[0 0],[1 0],[2 0],[3 0],[4 0]>: check pool content using pages
[test_assign2_2.c-Testing LRU_K page replacement-L134-12:05:21] OK: expected <[0 0],[1 0],[2 0],[5 0],[
4 0]> and was <[0 0],[1 0],[2 0],[5 0],[4 0]>: check pool content using pages
[test_assign2_2.c-Testing LRU_K page replacement-L134-12:05:21] OK: expected <[0 0],[1 0],[2 0],[5 0],[
6 0]> and was <[0 0],[1 0],[2 0],[5 0],[6 0]>: check pool content using pages
[test_assign2_2.c-Testing LRU_K page replacement-L134-12:05:21] OK: expected <[7 0],[1 0],[2 0],[5 0],[
6 0]> and was <[7 0],[1 0],[2 0],[5 0],[6 0]>: check pool content using pages
[test_assign2_2.c-Testing LRU_K page replacement-L134-12:05:21] OK: expected <[7 0],[1 0],[8 0],[5 0],[
6 0]> and was <[7 0],[1 0],[8 0],[5 0],[6 0]>: check pool content using pages
[test_assign2_2.c-Testing LRU_K page replacement-L134-12:05:21] OK: expected <[7 0],[9 0],[8 0],[5 0],[
6 0]> and was <[7 0],[9 0],[8 0],[5 0],[6 0]>: check pool content using pages
[test_assign2_2.c-Testing LRU_K page replacement-L138-12:05:21] OK: expected <0> and was <0>: check num
ber of write I/Os
[test_assign2_2.c-Testing LRU_K page replacement-L139-12:05:21] OK: expected <10> and was <10>: check n
umber of read I/Os
[test_assign2_2.c-Testing LRU_K page replacement-L146-12:05:21] OK: finished test
```

2. Testing CLOCK:

```
[test_assign2_2.c-Testing CLOCK page replacement-L192-16:11:18] OK: expected <[0 0],[-1 0],[-1
0],[-1 0],[-1 0]> and was <[0 0],[-1 0],[-1 0],[-1 0],[-1 0]>: check pool content reading in pa
ges
[test_assign2_2.c-Testing CLOCK page replacement-L192-16:11:18] OK: expected <[0 0],[1 0],[-1 0
],[-1 0],[-1 0]> and was <[0 0],[1 0],[-1 0],[-1 0],[-1 0]>: check pool content reading in page
s
[test_assign2_2.c-Testing CLOCK page replacement-L192-16:11:18] OK: expected <[0 0],[1 0],[2 0]
,[-1 0],[-1 0]> and was <[0 0],[1 0],[2 0],[-1 0],[-1 0]>: check pool content reading in pages
[test_assign2_2.c-Testing CLOCK page replacement-L192-16:11:18] OK: expected <[0 0],[1 0],[2 0]
,[3 0],[-1 0]> and was <[0 0],[1 0],[2 0],[3 0],[-1 0]>: check pool content reading in pages
[test_assign2_2.c-Testing CLOCK page replacement-L192-16:11:18] OK: expected <[0 0],[1 0],[2 0]
,[3 0],[4 0]> and was <[0 0],[1 0],[2 0],[3 0],[4 0]>: check pool content reading in pages
[test_assign2_2.c-Testing CLOCK page replacement-L200-16:11:18] OK: expected <[0 0],[1 0],[2 0]
,[3 0],[4 0]> and was <[0 0],[1 0],[2 0],[3 0],[4 0]>: check pool content using pages
[test_assign2_2.c-Testing CLOCK page replacement-L200-16:11:18] OK: expected <[0 0],[1 0],[2 0]
,[3 0],[4 0]> and was <[0 0],[1 0],[2 0],[3 0],[4 0]>: check pool content using pages
[test_assign2_2.c-Testing CLOCK page replacement-L200-16:11:18] OK: expected <[0 0],[1 0],[2 0]
,[3 0],[4 0]> and was <[0 0],[1 0],[2 0],[3 0],[4 0]>: check pool content using pages
[test_assign2_2.c-Testing CLOCK page replacement-L200-16:11:18] OK: expected <[0 0],[1 0],[2 0]
,[3 0],[4 0]> and was <[0 0],[1 0],[2 0],[3 0],[4 0]>: check pool content using pages
[test_assign2_2.c-Testing CLOCK page replacement-L200-16:11:18] OK: expected <[0 0],[1 0],[2 0]
,[3 0],[4 0]> and was <[0 0],[1 0],[2 0],[3 0],[4 0]>: check pool content using pages
[test_assign2_2.c-Testing CLOCK page replacement-L208-16:11:18] OK: expected <[0 0],[1 0],[2 0]
,[5 0],[4 0]> and was <[0 0],[1 0],[2 0],[5 0],[4 0]>: check pool content using pages
[test_assign2_2.c-Testing CLOCK page replacement-L208-16:11:18] OK: expected <[0 0],[1 0],[2 0]
,[5 0],[6 0]> and was <[0 0],[1 0],[2 0],[5 0],[6 0]>: check pool content using pages
[test_assign2_2.c-Testing CLOCK page replacement-L208-16:11:18] OK: expected <[0 0],[1 0],[2 0]
,[5 0],[6 0]> and was <[0 0],[1 0],[2 0],[5 0],[6 0]>: check pool content using pages
[test_assign2_2.c-Testing CLOCK page replacement-L208-16:11:18] OK: expected <[0 0],[1 0],[2 0]
,[5 0],[6 0]> and was <[0 0],[1 0],[2 0],[5 0],[6 0]>: check pool content using pages
[test_assign2_2.c-Testing CLOCK page replacement-L208-16:11:18] OK: expected <[0 0],[1 0],[2 0]
,[5 0],[6 0]> and was <[0 0],[1 0],[2 0],[5 0],[6 0]>: check pool content using pages
[test_assign2_2.c-Testing CLOCK page replacement-L212-16:11:18] OK: expected <0> and was <0>: c
heck number of write I/Os
[test_assign2_2.c-Testing CLOCK page replacement-L213-16:11:18] OK: expected <7> and was <7>: c
heck number of read I/Os
[test_assign2_2.c-Testing CLOCK page replacement-L220-16:11:18] OK: finished test
```

3. Testing ERRORS:

```
[test_assign2_2.c-ERROR TEST-L240-12:05:21] OK: expected an error and was RC <6>: try to pin page when
pool is full of pinned pages with fix-count > 0
[test_assign2_2.c-ERROR TEST-L246-12:05:21] OK: expected an error and was RC <6>: try to pin page with
negative page number
[test_assign2_2.c-ERROR TEST-L251-12:05:21] OK: expected an error and was RC <1>: try to init buffer po
ol for non existing page file
[test_assign2_2.c-ERROR TEST-L252-12:05:21] OK: expected an error and was RC <6>: shutdown buffer pool
that is not open
[test_assign2_2.c-ERROR TEST-L253-12:05:21] OK: expected an error and was RC <6>: flush buffer pool tha
t is not open
[test_assign2_2.c-ERROR TEST-L254-12:05:21] OK: expected an error and was RC <6>: pin page in buffer po
ol that is not open
[test_assign2_2.c-ERROR TEST-L259-12:05:21] OK: expected an error and was RC <6>: Try to unpin a page w
hich is not available in framelist.
[test_assign2_2.c-ERROR TEST-L260-12:05:21] OK: expected an error and was RC <6>: Try to forceflush a p
age which is not available in framelist.
[test_assign2_2.c-ERROR TEST-L261-12:05:21] OK: expected an error and was RC <6>: Try to markdirty a pa
ge which is not available in framelist.
[test_assign2_2.c-ERROR TEST-L269-12:05:21] OK: finished test
```