

```

1  /*
2  * DBMS Implementation
3  */
4
5  #ifndef BUFFEROPS_H
6  #define BUFFEROPS_H
7
8  #include <io.h>
9  #include <fcntl.h>
10 #include <sys/types.h>
11
12 #include "dbmsproj.h"
13 #include<iostream>
14
15
16 // empties a block
17 template<typename T> void emptyBlock(block_t<T> *buffer, unsigned int size) {
18     buffer->nreserved = 0;
19     buffer->entries.clear();
20     //printf("empty buffer block... \n");
21 };
22
23
24 // empties the whole buffer
25
26 template<typename T> void emptyBuffer(block_t<T> *buffer, unsigned int size) {
27     for (uint i = 0; i < size; i++) {
28         emptyBlock(buffer + i);
29         buffer[i].valid = true;
30     }
31     printf("empty buffer ...");
32 };
33
34 // opens filename for writing (append mode), and writes size blocks
35 // starting from pointer buffer
36
37 template<typename T> unsigned int writeBlocks(block_t<T> &relation, block_t<T> &buffer,  ➤
    unsigned int offset, unsigned int size) {
38     unsigned int writecounts = 0;
39
40     for (size_t i = 0; i < size; i++)
41     {
42         relation[i+offset] = buffer[i];
43         writecounts++;
44     }
45     printf("Write %d blocks into disk...\n", readcount);
46     return size;
47 };
48
49 // reads size blocks to buffer
50
51 template <typename T> unsigned int readBlocks(std::vector<block_t<T>>& relation,  ➤
    block_t<T>* buffer, unsigned int size) {
52     unsigned int readcount = 0;
53     for (unsigned int i = 0; i < size; i++)
54     {
55         if (i<relation.size())
56         {
57             buffer[i] = relation[offset + i];
58             readcount++;

```

```
59     }
60     else
61     {
62         break;
63     }
64 }
65 printf("loading %d blocks into buffer...\n",readcount);
66 return readcount;
67 };
68 template <typename T> unsigned int readBlocks(std::vector<block_t<T>>& relation,
        block_t<T>* buffer,unsigned int size,unsigned int offset) {
69     unsigned int readcount = 0;
70     for (unsigned int i = 0; i < size; i++)
71     {
72         if ((offset+i)<relation.size())
73         {
74             buffer[i] = relation[offset+i];
75             readcount++;
76         }
77         else
78         {
79             break;
80         }
81     }
82     printf("loading %d blocks into buffer...\n",readcount);
83     return readcount;
84 };
85 #endif
86
87
```