**Assignment 1**

CSEP 545

/* Start new transaction */
int start()
Returns: -1, if there is an active transaction
 otherwise
  tid > 0 (a new Transaction Identifier)
from disk into main memory and save a pointer to it in block */
int read(int blockAddr, int tid, &newBlock)
Returns: 0 if successful, -1 otherwise
/* tell B1 that transaction tid has updated a block at blockAddr */
int write(int blockAddr, int tid)
Returns: 0 if there is no cached entry for blockAddr
 otherwise
/* Try to commit transaction tid */
int commit(int tid)
Returns: 0 if successful, -1 otherwise
/* Abort transaction tid */
int abort(int tid)
Returns: 0

---

Start

lastTrans = 1;
... int start()
  if (lastTrans < 0) return -1
  else
    lastTrans = -(++lastTrans);
    return lastTrans;
}

---

Read

int read(int blockAddr, int tid, &newBlock)
  if (there is such a cache element e)
    Cache(e).tId = tid;
    block = &Cache(e).newBlock;
    return 0;
  else
    /* pick a cache entry e, where Cache(e).tId = 0.
       If there is no such entry, then return -1 */
    diskRead(blockAddr, &Cache(e).oldBlock);
    Cache(e).newBlock = Cache(e).oldBlock;
    block = &Cache(e).newBlock;
    Cache(e).tId = tid;
    return 0;
}

---

Write

/* A transaction should call
write(&Cache(e).newBlock, tid)
after it updates Cache(e).newBlock. */
int write(int blockAddr, int tid)
  { /* find the cache entry e for block blockAddr */
    if (there is no such entry) return 0
    else
      { Cache(e).tId = tid;
        return 1;
      }
  }

---

Commit

int commit(int tid)
  { /* for each cache entry e where Cache(e).tId == tid */
    status = diskWrite(Cache(e).blockAddr, &Cache(e).newBlock);
    Cache(e).tId = -tid;
    if (status == -1)
      { Abort(tid);
        return -1;
      }
    for (all cache entries e) set Cache(e).tId = 0;
    lastTrans = -lastTrans;
    return 0;
  }

---

Abort

int abort(int tid)
  { /* for all cache entries e, where Cache(e).tId == -tid */
    repeat
      status = diskWrite(Cache(e).blockAddr, &Cache(e).oldBlock)
      until (status == 0)
    for (all cache entries e)
      { Cache(e).tId = 0
       lastTrans = -lastTrans;
       return 0;
      }
  }
Questions

For each of the scenarios below, analyze whether transactions that use B1 satisfy the four ACID properties.

a. The OS and B1 never crash. Transactions run serially.
b. The operating system can crash, in which case main memory is lost. Transactions run serially.
c. The operating system and B1 never crash. Transactions run serially. But if DiskWrite(b, addr) returns 0 then it might have corrupted block b on disk.
   Variation of (c): If DiskWrite returns -1 then it might have corrupted block b on disk.
d. The operating system and B1 never crash. Two transactions are allowed to run concurrently.