Rešenja zadataka za
treći kolokvijum iz Operativnih sistema 1
Jun 2017.

1. **(10 poena)**

```
Byte* BlockIOCache::getBlock (BlkNo blk) {
  // Find the requested block in the cache and return it if present:
  int i, free = -1;
  for (i=0; i<CACHESIZE; i++) {
    if (this->flags[i]&F_VALID && this->cacheMap[i]==blk) {
      this->refCounter[i]++;
      return &this->cache[i];
    }
    if ((this->flags[i]&F_VALID==0 || this->refCounter[i]==0)
        && free==-1) free=i;
  }
  // The block is not in the cache, load it to the 'free' slot:
  if (free==-1) return 0; // A problem: there is no free space in the cache
  // Load the requested block:
  this->cacheMap[free] = blk;
  this->flags[free] = F_VALID;
  this->refCounter[free] = 1;
  ioRead(this->dev,blk,this->cache[free]);
  return this->cache[free];
}

void BlockIOCache::releaseBlock (Byte* buffer) {
  int i = (buffer-this->cache[0])/BLKSIZE;
  if (i<0 || i>=CACHESIZE) return; // Exception
  // If the block is dirty, write it to the device:
  if (this->flags[i]&F_VALID && this->flags[i]&F_DIRTY)
    ioWrite(this->dev,this->cacheMap[i],this->cache[i]);
  this->flags[i] &= ~F_DIRTY;
  this->refCounter[i]--;
}
7
```

2. **(10 poena)**

a)(5) `jane    chld    foo`

b)(5) `txt`

3. **(10 poena)**

```
void truncate (FCB* fcb) {
  if (fcb==0) return;
  unsigned long cur=fcb->head, next;
  while (cur) {
    next = fat[cur];
    fat[cur] = -1;
    cur = next;
  }
  fcb->head = 0;
  fcb->size = 0;
}
```