



linee codice e commenti	linee test SW	NOTE
5372	0	
3896	0	Non ammesso. Discutere il compito all'orale.

GiulianoLoMagro	
Fuori Spec	<ul style="list-style-type: none"> <li>- La directory entry di ogni file viene memorizzata insieme ai dati del file, precisamente nei primi 32 byte del cluster</li> <li>- Il pathname viene memorizzato nel nome del file e quindi la lunghezza del pathname e' limitata a 11 caratteri (es. mkdir /directory ; mkdir /dir1 ; mv /dir1 /directory/dir1 -&gt; ERRORE)</li> <li>- Le funzioni della libreria libufs.a (ufs_creat, ufs_open, ...) accedono direttamente al filesystem dev_ufs senza comunicare con il server. Solo la shell comunica con il server inviando i comandi ls, cat, touch, ...</li> </ul>
Relazione	<ul style="list-style-type: none"> <li>- Pag.4: "0x00000000 = partizione montata viene scritto da ufs_mount e 0x0000000F = partizione montata viene scritto da ufs_umount" ? 0x0 per partizione non montata (scritto da ufs_umount) e 0xF per partizione montata (scritto da ufs_mount).</li> <li>- Perche' in ufs_mke2fs questo campo viene messo pari ad 0x1 ?</li> <li>- Pag.4: "Ufs_creat è la funzione che emula la procedura creat dell' ANSI C" ??? Stessa cosa per la system call read (pag.5)</li> <li>- Pag.4: "Una volta trovata la locazione consona, la FAT e la casella di ClusterStatus che mappano quel cluster verranno poste rispettivamente a 0xffff e ad 1c" ? Perche' 0xffff e non 0xffffffff.</li> <li>- Stessa cosa a pag.5</li> <li>- Pag.4: "Dopo verranno copiate nel cluster scelto le informazioni che identificano il file (pathname,data di creazione,data di modifica ecc...),effettuando una scrittura,come da specifiche del progetto,sui primi 32 byte di questa locazione." ? Nelle specifiche del progetto c'e' scritto che queste informazioni (directory entry) sono memorizzate nella directory che contiene il file in questione.</li> <li>- Manca la descrizione delle strutture dati utilizzate</li> <li>- Non e' chiaro il meccanismo di lock utilizzato. ufs_fcntl mi sembra un wrapper a fcntl. Non viene implementata la funzionalita' aggiuntiva dei lock.</li> </ul>
Codice	<ul style="list-style-type: none"> <li>- Troppi if annidati. Il codice e' poco leggibile. Pochi commenti. Il codice non e' indentato.</li> <li>- Molte malloc senza controllo del valore di ritorno e senza free.</li> <li>- Vedi commenti SRU.</li> </ul>
Makefile	<ul style="list-style-type: none"> <li>- Non c'e' target install. Target all deve dipendere solo dagli eseguibili, non dai *.o.</li> <li>- Il makefile non utilizza le directory bin, obj, inc, lib e src come da specifica.</li> <li>- N.B. Il makefile originale e' in ORG</li> </ul>
Installaz	Non prevista
test	Non previsto
ufs_mke2fs	<ul style="list-style-type: none"> <li>- Ci sono print di debug. Tutti i valori scritti su dev_ufs vengono anche riletti e stampati su stdout.</li> <li>- Uso inutile di lseek. L'ultimo elemento della FAT viene sovrascritto dalla root-directory.</li> <li>- La dimensione del file e' corretta, ma l'algoritmo di allocazione dei cluster e' sbagliato!</li> <li>- Se consideriamo il fatto che viene sovrascritto l'ultimo elemento della FAT, tutti i test dell'eseguibile ufs_mke2fs falliscono</li> </ul>
ufs_mount, ufs_umount	<ul style="list-style-type: none"> <li>- ufs_mount non fa fork+exec, quindi non ritorna il prompt (ufsd va in fg). Vengono registrati 3 segnali, ma soltanto uno viene gestito dall'handler.</li> <li>- Il codice di ufs_umount contiene errori concettuali, fcntl(fd,...) close(fd) quando fd == -1.</li> <li>- ls non funziona sempre (vedi ls senza parametri), la UID stampata e' 200 mentre l'utente con cui ho fatto il test ha id 1001, la dimensione del file e' 0 anche se il file non e' vuoto. ls /dir1 stampa anche i file della directory /dir1/dir2</li> <li>- "cp -s" blocca il processo se il file e' piu' grande di un cluster. "cp -d" non funziona</li> <li>- mkdir permette di creare una directory con lo stesso nome di un file (es. touch /dir1/a ; mkdir /dir1/a ; ls /dir1)</li> <li>- tree mostra solo le directory (non stampa i file regolari), inoltre stampa le stesse informazioni di ls</li> <li>- touch permette di creare piu' file con lo stesso nome</li> </ul>
libufs.a: GENERAL	Vedi commenti per singolo test group.
UFS.CD	Il test UFS.CD.02b si blocca sull'istruzione "while ((fd=open(wserver,O_RDONLY))<0)" della funzione ufs_disconnect() (file: libufs.c). Sembra che nessun altro processo abbia aperto la FIFO in scrittura.
UFS.FCRR	18 Test, 7 errori. Commentati i test: UFS.FCRR.10a/b/c/d e UFS.FCRR.30. Tot. 23 test, 11 passed: 11*10/23 = 4.7
UFS.FOC	11 Test, 2 errori. Commentati i test: UFS.FOC.10a/b/c e UFS.FOC.40c. Tot. 15 test, 9 passed: 9*10/15 = 6
UFS.FRWL	Segmentation fault alla prima write
UFS.FST	8 Test, 1 errore. Commentati 17 test. Su 25 test, 7 passed: 7*10/25 = 2.8
UFS.DMRR	29 Test, 11 errori. Commentati 3 test. Tot. 32 test, 18 passed: 18*10/32 = 5.6
UFS.DOC	7 Test, 2 errori. Commentati 3 test. Tot. 10 test, 5 passed: 5*10/10 = 5
UFS.DRD	La maggior parte dei test non funziona
UFS.DST	La maggior parte dei test non funziona
Penalizzazioni	Il codice e' stato corretto in qualche parte (eliminate printf in ufs_mke2fs, ...).

X

Cavallo-Tognazzi	
Fuori spec	file "Aouthours", manca README e dir test
Relazione	<p>Z.0: "... e provare la terminazioni di eventuali server ancora attivi ": cosa vuol dire?</p> <p>Diverse imprecisioni, errori e termini colloquiali: ("inizializzazione", "killa", "in specifico", "modo molto intuitiva", "come installatore del progetto", "preoccupera' a configurare", "Comunque di default il numero è dimensione dei cluster è dimensione a 1024", ...)</p> <p>Perche' si afferma che ufsh fa la connect/disconnect per ogni comando?</p> <p>La relazione spiega troppo cio' che e' gia' scritto nelle spec, troppo poco l'organizzazione interna delle strutture dati a supporto dei vari algoritmi.</p>
Codice	<p>Struttura standard, codice abbastanza ordinato con commenti. Spesso troppi if annidati.</p> <p>Manca gestione errore su malloc, NON ESISTONO FREE!</p> <p>Vengono allocate variabili di tipi base!</p> <p>Dove sono tutti i prototipi e le typedef di libufs.a???</p>
Makefile	<p>Target all deve dipendere solo dagli eseguibili, non dai *.o.</p> <p>La dip da obj non per muovere i file e' una prassi discutibile.</p> <p>Ogni *.o deve dipendere dal *.c. Il link di ogni eseguibile deve utilizzare il *.o non il *.c, l'eseguibile non dipende da .h, ma dai *.o linkati.</p> <p>Alcuni warning in compilazione: "assignment makes pointer from integer without a cast."</p> <p>Non c'e' target clean e install.</p> <p>Il makefile ricompila sempre!</p>
Installaz	Non e' prevista.
test SW	Non esiste.
ufs_mke2fs	Funziona solo default, errore nel test dimCluster e tutti i confronti tra stringhe errati (ma questi sono errori da Prog 1 e 2 !
ufs_mount, ufs_umount	<p>ufs_mount non fa fork+exec, quindi non ritorna il prompt (ufsd va in fg).</p> <p>ufs_umount usa kill (killall): errato, non si devono usare comandi esterni, solo syscall. Comunque inserisco pkill con execip: ufsd viene terminato ma rimane una FIFO.</p>
ufsh	Is non e' riconosciuto, non e' possibile vedere il risultato di un mkdir, cp da sempre errore, impossibile fare il test!
libufs.a: GENERAL	Impossibile procedere con i test! Si blocca sulla prima creat o mkdir, non si puo' procedere!!!
UFS.CD	01a e 02b ritornano un ret code sbagliato, rimangono FIFO dedicate in /tmp.
UFS.FCRR	Si blocca sulla prima creat 01a.
UFS.FOC	
UFS.FRWL	
UFS.FST	
UFS.DMRR	
UFS.DOC	
UFS.DRD	
UFS.DST	