

Soluzioni della Prova Scritta di Sistemi di Elaborazione Ingegneria delle Telecomunicazioni

Ing. G. Lettieri, Ing. A. Vecchio

1 Febbraio 2007

1. (a) Il programma principale riceve due argomenti da riga di comando: il primo rappresenta il nome di un comando, il secondo il nome di un file. Il programma principale apre in lettura il file che ha nome `arg[2]` e legge il suo contenuto: ogni stringa viene memorizzata all'interno di una riga di una matrice di caratteri. La funzione `f1` esegue un ciclo in cui richiama la funzione `f2` passandole come argomenti il nome del comando e una delle stringhe contenute nella matrice. La funzione `f2` esegue il comando indicato dalla prima stringa in un nuovo processo, passandogli come argomento la seconda stringa ricevuta.

In sintesi, il programma principale esegue ripetutamente il comando indicato da `args[1]` passandogli come argomenti le stringhe contenute nel file di nome `args[2]`.

- (b) Una possibile traduzione è la seguente:

```
.text                                pushl 8(%ebp)
.global f2                            call  execlp
f2:  pushl %ebp                        addl  $16, %esp
      movl  %esp, %ebp                jmp   fine
      pushl %eax                       else: pushl $0
      call  fork                       call  wait
      cmpl  $0, %eax                   addl  $4, %esp
      jne  else                         fine: popl %eax
      pushl $0                          leave
      pushl 12(%ebp)                    ret
      pushl 8(%ebp)
```

2. (a)

```
#include <sys/types.h>
#include <sys/stat.h>
#include <fcntl.h>
#include <stdio.h>
#include <unistd.h>

#define MAX_NAME 256
#define BUF_SIZE 1024

int main(int argc, char* argv[])
{
    char name[MAX_NAME];
    char buf[BUF_SIZE];
    int nlines, count, nfile;
    FILE *fin, *fout;
    struct stat st;

    if (argc < 3) {
```

```

        fprintf(stderr, "Uso: %s <linee> <file>\n", argv[0]);
        exit(1);
    }

    if (sscanf(argv[1], "%d", &nlines) != 1 || nlines <= 0) {
        fprintf(stderr, "Il primo argomento deve essere un intero > 0\n");
        exit(1);
    }

    if (stat(argv[2], &st) < 0) {
        perror(argv[2]);
        exit(1);
    }

    if (!S_ISREG(st.st_mode)) {
        fprintf(stderr, "%s: non e' un file regolare\n", argv[2]);
        exit(1);
    }

    if ( (fin = fopen(argv[2], "r")) == NULL) {
        perror(argv[2]);
        exit(1);
    }

    nfile = 0;
    fout = NULL;
    count = nlines;
    while (fgets(buf, BUF_SIZE, fin) != NULL) {
        if (++count > nlines) {
            if (fout) fclose(fout);
            sprintf(name, "%s-%d", argv[2], nfile);
            if ( (fout = fopen(name, "w")) == NULL) {
                perror(name);
                exit(1);
            }
            count = 1;
            nfile++;
        }

        if (fputs(buf, fout) == EOF) {
            perror(name);
            exit(1);
        }
    }
    return 0;
}

```

```

(b) #include <sys/types.h>
    #include <sys/stat.h>
    #include <fcntl.h>
    #include <stdio.h>

```

```

#include <unistd.h>

#define MAX_NAME 256
#define BUF_SIZE 1024

int main(int argc, char* argv[])
{
    int i, fd;
    int nproc = 0;

    if (argc < 3) {
        fprintf(stderr, "Uso: %s <log> <file>...\n", argv[0]);
        exit(1);
    }

    if ( (fd = open(argv[1], O_WRONLY|O_CREAT|O_APPEND, 0666)) < 0)
        perror(argv[1]);

    for (i = 2; i < argc; i++) {

        switch (fork()) {
        case -1:
            perror(argv[0]);
            break;
        case 0:
            if (fd >= 0) {
                close(2);
                dup(fd);
                close(fd);
            }
            execl("split", "split", argv[i], NULL);
            perror("split");
            exit(1);
        default:
            nproc++;
            break;
        }

    }

    for (i = 0; i < nproc; i++)
        wait(0);
    return 0;
}

```