

# Soluzioni della Prova Scritta di Sistemi di Elaborazione Ingegneria delle Telecomunicazioni

Ing. G. Lettieri, Ing. A. Vecchio

16 Febbraio 2007

1. (a) La funzione `f1()` prende come argomenti una vettore di interi e la sua lunghezza. Passa gli elementi del vettore a coppie alla funzione `f2()` e stampa (tramite la funzione `f3()`) il valore ricevuto. La funzione `f2()` prende come argomenti due interi e restituisce il minore tra i due. Complessivamente, il programma interpreta il contenuto del file `input.txt` come una sequenza di coppie di interi e, per ogni coppia, stampa il valore minimo.
- (b) Una possibile traduzione è la seguente:

```
.text                                pushl (%ebx, %edi, 4)
.global f1                            movl %esi, %edi
f1:  pushl %ebp                        shll %edi
      movl %esp, %ebp                 pushl (%ebx, %edi, 4)
      # uso %esi come i               call f2
      pushl %eax                      addl $8, %esp
      pushl %ebx                      pushl %eax
      pushl %edi                      call f3
      pushl %esi                      addl $4, %eax
                                      incl %esi
                                      jmp  ciclo
      movl 8(%ebp), %ebx              fine:
      shr 12(%ebp)
      movl $0, %esi                  popl %esi
ciclo:                                popl %edi
      cmpl 12(%ebp), %esi            popl %ebx
      jge fine                       popl %eax
      movl %esi, %edi                leave
      shll %edi                      ret
      incl %edi
```

2. (a) `#include <stdio.h>`

```
#define MAX_LINE 1024

int main(int argc, char* argv[])
{
    char buf1[MAX_LINE];
    char buf2[MAX_LINE];
    char *last, *curr, *tmp;

    if (argc != 1) {
        fprintf(stderr, "Uso: %s\n", argv[0]);
        exit(1);
    }
}
```

```

curr = buf2;
last = fgets(buf1, MAX_LINE, stdin);
while (last) {
    int count = 1;

    while ( (curr = fgets(curr, MAX_LINE, stdin))
            && strcmp(curr, last) == 0)
        count++;

    printf("%d\t%s", count, last);

    tmp = last;
    last = curr;
    curr = tmp;
}

return 0;
}
(b) #include <unistd.h>
#include <stdio.h>
#include <fcntl.h>

#define MAX_FILES 10

int main(int argc, char* argv[])
{
    int p[2], i, j;
    char* s_argv[MAX_FILES + 2];

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

    if (argc > 2 + MAX_FILES) {
        fprintf(stderr, "troppi argomenti\n");
        exit(1);
    }

    if (pipe(p) < 0) {
        perror(argv[0]);
        exit(1);
    }

    switch (fork()) {
    case 0:
        close(1);
        dup(p[1]);
        close(p[0]);
        close(p[1]);
        s_argv[0] = "sort";
        j = 1;

```

```

        for (i = 2; i < argc; i++)
            s_argv[j++] = argv[i];
        s_argv[j] = NULL;
        execvp("sort", s_argv);
        perror("sort");
        exit(1);
    case -1:
        perror(argv[0]);
        exit(1);
    }
    switch (fork()) {
    case 0:
        close(0);
        dup(p[0]);
        close(p[0]);
        close(p[1]);
        close(1);
        if (open(argv[1], O_CREAT|O_TRUNC|O_WRONLY, 0666) < 0) {
            perror(argv[1]);
            exit(1);
        }
        execl("ripetizioni", "ripetizioni", NULL);
        perror("ripetizioni");
        exit(1);
    case -1:
        perror(argv[0]);
        exit(1);
    }
    close(p[0]);
    close(p[1]);

    wait(0);
    wait(0);

    return 0;
}

```