

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 stringa e un carattere e restituisce al chiamante il numero di occorrenze del carattere all'interno della stringa. La funzione `f2()` prende come argomenti due stringhe, `a` e `b`, e calcola la somma delle occorrenze dei singoli caratteri di `b` all'interno di `a`. Il programma principale calcola e stampa a video la somma delle occorrenze dei singoli caratteri contenuti in `argv[1]` per ognuna delle stringhe contenute nel file `input.txt`.

- (b) Una possibile traduzione è la seguente:

```
.text                                cmpb  $0x00, (%ecx, %edi)
.global f2                            je    avanti
f2:  pushl %ebp                        pushl (%ecx, %edi)
      movl %esp, %ebp                 pushl 8(%ebp)
      subl $4, %esp                   call  f1
      #uso %edi come i                 addl  $8, %esp
      #uso -4(%ebp) come r             addl  %eax, -4(%ebp)
      #8(%ebp) e' a                   incl  %edi
      #12(%ebp) e' b                  jmp   while
      pushl %ecx                       avanti:
      pushl %edi                       movl  -4(%ebp), %eax
      movl  $0, -4(%ebp)                popl  %edi
      movl  $0, %edi                    popl  %ecx
      movl  12(%ebp), %ecx              leave
while:                                  ret
```

2. (a) `#include <stdio.h>`
`#include <unistd.h>`
`#include <sys/stat.h>`
`#include <fcntl.h>`
- `#define MAXLINE 80`
- ```
int main(int argc, char* argv[]) {
 FILE *f;
 char buf[MAXLINE];

 if (argc != 2) {
 fprintf(stderr, "Uso: %s <nome file>\n", argv[0]);
 exit(1);
 }

 if (!(f = fopen(argv[1], "r"))) {
 perror(argv[1]);
 exit(1);
 }
}
```

```

 }

 while (fgets(buf, MAXLINE, stdin)) {
 int n, m = -1;

 if (sscanf(buf, "%d\n", &n) != 1 || n < 0) {
 fprintf(stderr, "ingresso errato\n");
 continue;
 }

 fseek(f, 0, SEEK_SET);
 do {
 if (!fgets(buf, MAXLINE, f)) {
 fprintf(stderr, "fine file\n");
 break;
 }
 m++;
 } while (m < n);
 if (m == n)
 printf("%s", buf);
 }

 return 0;
}

(b) #include <unistd.h>
#include <stdio.h>

int main(int argc, char* argv[])
{
 int p[2];
 int n;

 if (argc != 3) {
 fprintf(stderr, "Uso: %s <nome file> <numero linee>\n", argv[0]);
 exit(1);
 }

 if (sscanf(argv[2], "%d", &n) != 1 || n < 0) {
 fprintf(stderr, "secondo parametro errato\n");
 exit(1);
 }

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

 switch (fork()) {
 case -1:
 perror(argv[0]);
 exit(1);
 case 0:

```

```

 close(0);
 dup(p[0]);
 close(p[0]);
 close(p[1]);
 execl("lines", "lines", argv[1], NULL);
 perror("lines");
 exit(1);
default:
 break;
}
switch (fork()) {
case -1:
 perror(argv[0]);
 exit(1);
case 0:
 close(1);
 dup(p[1]);
 close(p[0]);
 close(p[1]);

 for (n--; n >=0; n--)
 printf("%d\n", n);
default:
 break;
}

close(p[0]);
close(p[1]);
wait(0);
wait(0);
return 0;
}

```