

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

Ing. G. Lettieri, Ing. A. Vecchio

18 settembre 2009

1. (a) Il programma prende in ingresso, da riga di comando, due lettere (chiamiamole *a* e *b*). Quindi, ciclicamente, legge una riga dall'ingresso standard, sostituisce tutte le occorrenze di *a* con *b* all'interno di tale riga, e infine la stampa sull'uscita standard. Il programma termina quando legge una riga uguale a ".".
- (b) Una possibile traduzione è la seguente:

<pre>.data str_punto: .asciz "."  .text .global f1 f1:     pushl %ebp     movl  %esp, %ebp     subl  MAXL, %esp     movl  %esp, %ebx ciclo:     pushl %ebx     call  gets     addl  \$4, %esp     pushl \$str_punto     pushl %ebx     call  strcmp</pre>	<pre>addl  \$8, %esp cmpl  \$0, %eax je    fine pushl 12(%ebp) pushl 8(%ebp) push  %ebx call  f2 addl  \$12, %esp pushl %ebx call  puts addl  \$4, %esp jmp   ciclo fine: leave ret</pre>
---	---

2. (a) 

```
#include <stdio.h>
#include <string.h>

#define MAXLINE 1024

void conta(FILE* f) {
    char buf[MAXLINE];
    int conta = 0;

    while ( fgets(buf, MAXLINE, f) )
        conta++;
    printf("%d\n", conta);
}

int main(int argc, char* argv[])
{
    FILE *in = NULL;
    int i;
```

```

    if (argc == 1) {
        conta(stdin);
        return 0;
    }
    for (i = 1; i < argc; i++) {
        if (argc > 2)
            printf("%s\t", argv[i]);
        if (strcmp(argv[i], "-") == 0)
            conta(stdin);
        else {
            if ( !(in = fopen(argv[i], "r")) ) {
                perror(argv[i]);
                continue;
            }
            conta(in);
            fclose(in);
        }
    }
    return 0;
}

```

(b) #include <stdio.h>  
#include <string.h>  
#include <stdlib.h>

```

#define MAXLINE 1024

int estrai(int col, char* buf)
{
    int i = 0, n = 0;
    while (buf[i] && n < col) {
        if (buf[i] == '\t')
            n++;
        i++;
    }
    return atoi(&buf[i]);
}

int main(int argc, char *argv[])
{
    char buf[MAXLINE];
    int tot, col;
    if (argc != 2 || sscanf(argv[1], "%d", &col) != 1) {
        fprintf(stderr, "Uso: %s <col>\n", argv[0]);
        exit(1);
    }
    if (col < 0) {
        fprintf(stderr, "col deve essere maggiore o uguale a 0\n");
        exit(1);
    }
    tot = 0;
    while (fgets(buf, MAXLINE, stdin))

```

```

        tot += estrai(col, buf);
    printf("%d\n", tot);
}
(c) #include <stdio.h>
#include <stdlib.h>
#include <unistd.h>

int main(int argc, char* argv[])
{
    int pf[2], i;

    if (argc < 2) {
        fprintf(stderr, "Uso: %s file1 file2...\n", argv[0]);
        exit(1);
    }

    if (pipe(pf) < 0) {
        perror(argv[0]);
        exit(1);
    }
    switch (fork()) {
    case -1:
        perror(argv[0]);
        exit(1);
    case 0:
        close(1);
        dup(pf[1]);
        close(pf[0]);
        close(pf[1]);
        execv("linee", argv);
        perror("linee"),
        exit(1);
    default:
        break;
    }
    switch (fork()) {
    case -1:
        perror(argv[0]);
        exit(1);
    case 0:
        close(0);
        dup(pf[0]);
        close(pf[0]);
        close(pf[1]);
        execl("somma", "somma", (argc < 3 ? "0" : "1"), NULL);
        perror("somma");
        exit(1);
    default:
        break;
    }
    close(pf[0]);
    close(pf[1]);
}

```

```
    wait(NULL);  
    wait(NULL);  
    return 0;  
}
```