

Corso di Sistemi Operativi  
Proff. Abate, Rescigno – Anno Acc. 2018/19  
Appello I (**Laboratorio**) del 07 gennaio 2019

Cognome e Nome \_\_\_\_\_ Matricola \_\_\_\_\_/\_\_\_\_\_

Riservato per la correzione

1	2	Totale
/22	/8	/30

**NB. non è necessario inserire direttive `#include`; i commenti non sono necessari, ma potrebbero essere utili in caso di errore;**

Esercizio 1 (22 pt)

Scrivere un programma in linguaggio C che in sequenza:

- crei 2 pipe e un figlio (le 2 pipe saranno utilizzate per una comunicazione bidirezionale tra padre e figlio);
- dopo la creazione del figlio prenda in input dall'utente un nome di directory ed un nome di file;
- invii al figlio il nome di directory ed il nome di file utilizzando la prima pipe;
- faccia controllare al figlio se il file è contenuto nella directory e comunichi tale informazione al padre utilizzando la seconda pipe;
- in caso affermativo comunichi al padre anche il numero di stringhe contenute nel file;
- faccia stampare al padre le informazioni ricevute dal figlio.



Esercizio 2 (8 pt)

Si assuma di compilare ed eseguire il seguente programma

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/types.h>
#include <sys/wait.h>

void exit1(void);
void exit2(void);

int main(void) {
    int pid;
    char array[]="Dopo\n";
    atexit(exit1);

    printf("Prima");

    if((pid=fork())<0)
        exit(1);

    if(pid==0) {
        atexit(exit2);
        printf("Sono il figlio\n");
        write(1,array,5);
        _exit(0);
    }
    wait(NULL);
    printf("Sono il padre\n");
    write(1,array,5);
    printf("Fine esecuzione");
    _exit(0);
}

static void exit1(void) {
    printf("Sono il primo handler");
}
static void exit2(void) {
    printf("Sono il secondo handler");
}
```

Spiegare quale sarà l'output del programma, distinguendo tra processo padre e processo figlio, giustificando in maniera esaustiva la risposta.

