



Cognome	Nome	Matricola	Voto: ... /30
----------------	-------------	------------------	----------------------

Quesito:	1	2	3	4	5	6	Tot.
Max:	6	6	6	5	4	3	30
Punti:							

Istruzioni:

- non è possibile consultare libri, appunti, la calcolatrice o qualsiasi dispositivo elettronico, né comunicare;
- si può scrivere con qualsiasi colore, anche a matita, ad eccezione del **rosso**.
- tempo a disposizione: 2h 00m

Stile del codice C:

- non è necessario inserire direttive `#include`;
- i commenti non sono necessari, ma potrebbero essere utili nel caso di errore;

Quesito 1 (6 punti)

Punteggio ottenuto: ... /6

Scrivere un **sottoprogramma** chiamato `eliminaduplicati` che ricevuta in ingresso una lista la restituisce dopo aver eliminato eventuali duplicati, mantenendo per i valori lo stesso ordine in cui compaiono nella lista inizialmente.

Esempio:

Ingresso: 2 6 3 7 3 8 2 1 0 3 6 8 3 7
Uscita: 2 6 3 7 8 1 0

Quesito 2 (6 punti)

Punteggio ottenuto: ... /6

Scrivere un **sottoprogramma** che ricevuta in ingresso una stringa, stampa a video tutte le possibili combinazioni di caratteri adiacenti nella stringa (ognuna su una riga diversa) di dimensione da 1 fino alla lunghezza della stringa stessa.

Esempio:

Ingresso: XYZ
Uscita: X
 Y
 Z
 XY
 YZ

Quesito 3 (6 punti)

Punteggio ottenuto: ... /6

Scrivere un programma che chiede all'utente di inserire un numero n intero incluso nell'intervallo $[1, N]$ e visualizza tutte le possibili combinazioni di n elementi, ciascuno dei quali può assumere valore '+' oppure '-'. N è definito in una opportuna `#define`. Se l'utente inserisce un valore n non valido, il valore viene chiesto nuovamente. L'ordine con cui vengono visualizzate le combinazioni non è importante.

Esempio:

Ingresso: 3
Uscita: +++
 ++-
 +-+
 +--
 -++
 -+-
 ---+

Quesito 4 (5 punti)

Punteggio ottenuto: .../5

Cosa fa il programma qui di seguito riportato (dare una specifica astratta e non descrivere il codice)? Cosa viene visualizzato nel caso in cui l'utente inserisca abracadabra e cara?

```
#define LBIG 80
#define LSMALL 50

int main (int argc, char * argv[])
{
    char sequence[LBIG+1], pattern[LSMALL+1];
    int lseq, lpatt, mindiff, diff;
    int i, off, j, found;

    scanf("%s%s", sequence, pattern);

    lseq = strlen(sequence);
    lpatt = strlen(pattern);

    for(i = 0, j = 0, found = 0, mindiff = lpatt; i + lpatt <= lseq; i++){
        for(off = 0, diff = 0; off < lpatt; off++){
            if(sequence[i+off] != pattern[off])
                diff++;
            if(diff < mindiff) {
                found = i;
                mindiff = diff;
            }
        }
        printf("%s\n", sequence);
        for(j = 0; j < found; j++)
            printf(" ");
        for(i = 0; i < lpatt; i++)
            if(sequence[i+found] != pattern[i])
                printf("-");
            else
                printf("%c", pattern[i]);
        printf("\n");

    return 0;
}
```

Quesito 5 (4 punti)

Punteggio ottenuto: .../4

Cosa visualizza il programma qui riportato se l'utente fornisce in ingresso il valore 4? È possibile rispondere alla domanda oppure la visualizzazione non può essere deterministicamente specificata?

```
int main(int argc, char * argv[])
{
    int status;
    int n, i;

    scanf("%d", &n);

    for(i = 0; i < n; i++)
        if(fork() == 0) {
            printf("figlio %d\n", i);
            sleep(i+1);
            exit(0);
        } else {
            wait(&status);
            printf("figlio %d terminato\n", i);
        }

    return 0;
}
```

Quesito 6 (3 punti)

Punteggio ottenuto: .../3

Illustrare l'organizzazione logica del bus di sistema di un calcolatore in relazione al modello di Von Neumann, indicando il ruolo di ogni sua parte.