

**Laboratorio di Informatica**

Università di Catania  
Dipartimento di Ingegneria Informatica e delle Telecomunicazioni  
Corso di Laurea in Ingegneria Informatica  
N.O., A.A. 2002/03

***Laboratorio di Informatica***

*Ing. Ferdinando Portuese*

## **1<sup>a</sup> Prova intermedia**

Vengono proposti 3 gruppi di Domande:

- |  |               |
|--|---------------|
| ✓ <b>Gruppo A</b> : domande di carattere <u>Generale</u> (verifica conoscenze acquisite) | Max. 16 punti |
| ✓ <b>Gruppo B</b> : domande di <u>Analisi</u>  | Max. 10 punti |
| ✓ <b>Gruppo C</b> : domande di <u>Sintesi</u>  | Max. 4 punti  |

- Accanto ad ogni domanda è indicato il relativo punteggio massimo attribuibile.
- La sufficienza alla prova si ottiene se il punteggio totalizzato è almeno **18**.
- La sufficienza in questa prova è necessaria per l'ammissione alla seconda prova (pratica).
- Tra le risposte proposte per ciascuna domanda più di una potrebbe essere esatta.
- Utilizzate la riga “**Eventuale commento** \_\_\_\_\_” per motivare eventualmente la risposta.
- Il tempo a disposizione è di 2 ore.
- Non è concesso l'uso di testi o appunti.

---

Studente : \_\_\_\_\_ Matricola \_\_\_\_\_

Anno di corso : \_\_\_\_\_

**Firma**

\_\_\_\_\_

## Laboratorio di Informatica

## Gruppo A

**A1** – (0,5 punti) Supponiamo di dover assegnare due strutture identiche S1 ed S2 come sotto definite; quali delle seguenti istruzioni **non è consentita** per effettuare l'assegnamento ?

```
struct prova { int  A;
               float B;
             }
struct prova S1;
struct prova S2;
```

- ☐ S1 = S2;  
☐ S1 == S2;  
☐ S1.A = S2.A; S1.B = S2.B;

*Eventuale commento* \_\_\_\_\_

**A2** – (0,5 punti) Data la seguente definizione di struttura, per indicare che si tratta di una persona coniugata occorre scrivere:

```
struct Persona {
    char nome[25];
    char indir[25];
    char CodF[16];
    char coniugato;
} Anagrafica;
```

- ☐ Persona.coniugato = 'S';  
☐ Anagrafica.coniugato = 'S';  
☐ coniugato.Anagrafica = 'S';  
☐ Persona.Anagrafica.coniugato = 'S';

*Eventuale commento* \_\_\_\_\_

**A3** - (2,5 punti) Siano date le seguenti dichiarazioni di strutture e definizione di variabile:

```
struct canzone{
    char titolo[30];
    long int durata; /* espressa in secondi */
};
struct CD {
    char titolo[30];
    int  numCanzoni;
    struct canzone canzoni[25];
    char autore[30];
    float prezzo;
};
```

```
struct CD miacollezione[30];
```

Scrivere le istruzioni per

- 1) Assegnare al primo CD di miacollezione il prezzo di €13 (assegnare 13 al campo prezzo).
- 2) Assegnare al campo numCanzoni del quarto CD di miacollezione il valore 15.
- 3) Assegnare alla terza canzone del secondo CD di miacollezione la durata di 2min e 30sec.
- 4) Assegnare al primo carattere del titolo della seconda canzone del quinto CD il valore 'L'
- 5) Assegnare al primo carattere del titolo del terzo CD il valore 'd'.

## Laboratorio di Informatica

**A4 - ( 1 punto)** Se s1 ed s2 sono due stringhe `char s1[25]; char s2[25];` e si scrive `s1 = s2;` cosa succede?

- ☐ Tutto il contenuto di s2 viene copiato in s1  
☐ Si ottiene un errore di compilazione  
☐ Il primo elemento di s2 viene copiato nel primo elemento di s1

*Eventuale commento* \_\_\_\_\_

**A5 - ( 0,5 punti)** Sia dato il seguente frammento di programma:

```
int *fp();
```

- ☐ è sintatticamente scorretto  
☐ è la dichiarazione di una funzione senza parametri che restituisce un puntatore a intero  
☐ è una dichiarazione inammissibile in C

**A6 - ( 1 punto)** Sia dato il seguente frammento di programma:

```
int funz1( );  
int funz2[ ];
```

- ☐ è sintatticamente scorretto  
☐ sono due dichiarazioni di funzioni senza parametri che restituiscono un intero  
☐ sono due dichiarazioni: la prima dichiara funz1 funzione senza parametri che restituisce un intero, la seconda dichiara funz2 array di interi, in numero da stabilire a tempo di esecuzione.

*Eventuale commento* \_\_\_\_\_

**A7 - (2 punti)** Siano **x** e **y** interi e **z** reale. Siano inoltre **p** e **q** puntatori a interi ed **r** puntatore a reale. Indicare in tabella i cambiamenti operati da ciascuna istruzione. Nel caso in cui l'istruzione si ritiene errata sintatticamente, riportare nella colonna errore un \*.

Istruzione	x	y	z	*p	*q	*r	errore
x = 5; y = 10;	5	10					
z = x;							
p = &y;							
q = p;							
r = q;							
*r = 12;							
z = z + *r;							
p = &(x+1);							
*q = *z + 1;							

*Eventuale commento* \_\_\_\_\_

## Laboratorio di Informatica

**A8 - (2,5 punti)** Siano **X** e **Y** due vettori di interi. Siano inoltre **p** e **q** puntatori a interi.

Dopo aver segnato le istruzioni che si ritengono errate (con ☒) seguire l'evoluzione delle istruzioni (ignorando quelle errate) e riportare i contenuti dei due vettori:

<pre>int n = 2; int X[5] = {1, 2, 3, 4, 5 }; int Y[5] = {6, 7, 8, 9, 10};</pre>	<div style="display: flex; flex-direction: column; gap: 5px;"> <div><input type="checkbox"/> p = X ;</div> <div><input type="checkbox"/> *(p+n) = Y[n] ;</div> <div><input type="checkbox"/> Y = p ;</div> <div><input type="checkbox"/> q = &amp;Y[4];</div> <div><input type="checkbox"/> n = *q;</div> <div><input type="checkbox"/> *(q-2) = n;</div> <div><input type="checkbox"/> p = Y;</div> <div><input type="checkbox"/> *p = n;</div> </div>
---	---

  

Valori a fine codice:					
<div style="display: flex; align-items: flex-start;"> <div style="margin-right: 10px;">X</div> <table border="1" style="border-collapse: collapse; text-align: center;"> <tr><td style="height: 20px; width: 20px;"></td><td style="width: 20px;"></td><td style="width: 20px;"></td><td style="width: 20px;"></td><td style="width: 20px;"></td></tr> </table> </div>					
<div style="display: flex; align-items: flex-start;"> <div style="margin-right: 10px;">Y</div> <table border="1" style="border-collapse: collapse; text-align: center;"> <tr><td style="height: 20px; width: 20px;"></td><td style="width: 20px;"></td><td style="width: 20px;"></td><td style="width: 20px;"></td><td style="width: 20px;"></td></tr> </table> </div>					

**A9 - (0,5 punti)** Di che tipo è la variabile **f** restituita dalla funzione **fopen** qui di seguito riportata (si scriva anche la dichiarazione):

**f = fopen("dati.txt", "r");**

**Tipo restituito:** \_\_\_\_\_

**Dichiarazione:** \_\_\_\_\_

**A10 - (1 punto)** Dato il file "dati.txt" di seguito riportato e contenente due righe di testo, se **ch** è di tipo carattere e **fp** è un puntatore a file, quante volte cicla il costrutto **do ... while** contenuto nel frammento di codice?

<pre>fp = fopen("dati.txt", "r"); do  fscanf(fp, "%c", &amp;ch); while (!feof(fp)); fclose(fp);</pre>	<div style="display: flex; align-items: center;"> <div style="margin-right: 10px;">dati.txt</div> <table border="1" style="border-collapse: collapse; text-align: center;"> <tr><td style="height: 20px; width: 40px;"></td></tr> <tr><td style="height: 20px; width: 40px;">pippo</td></tr> <tr><td style="height: 20px; width: 40px;">pluto</td></tr> </table> </div>		pippo	pluto
pippo				
pluto				

**Risposta:** \_\_\_\_\_

**A11 - (1 punto)** Data la seguente funzione dove il passaggio di parametri avviene per valore, trasformarla in una funzione dove il passaggio è per indirizzo :

<pre>int Incremento(int a, int b) {     int somma;     a++;     b++;     somma = a + b;     return somma; }</pre>	<pre>int Incremento(_____, _____) {     int somma;     _____;     _____;     somma = _____;     return somma; }</pre>
---	---

## Laboratorio di Informatica

A12 – (1 punto) Sia dato il seguente programma:

```
#include<stdio.h>

void stampa(int b);
int f(int c);

main () {
    int a=3;
    stampa(a);
}

void stampa(int b)
{printf("%d",f(b));}

int f(int c){return (c*c);}
```

- ☐ il programma contiene un errore sintattico
- ☐ il programma contiene più di un errore sintattico
- ☐ l'esecuzione del programma provoca la stampa di 9

Eventuale commento \_\_\_\_\_

A13 – (1 punto) Sia dato il seguente programma:

```
#include <stdio.h>
#define N 10

main(){
    int k, v[N] = {0,1,2,3,4,5,6,7,8,9};
    for (k=1; k<=N; k++) printf("%d ",*(v+k));
    return;
}
```

esso è corretto sia dal punto di vista sintattico che logico?

☐ Si

☐ No

*Perché ?* \_\_\_\_\_

A14 - (1 punto) Sia dato il seguente frammento di programma:

```
if ( (char *) malloc(sizeof(int)) <= (char *)malloc(sizeof(long int)) )
    printf("%d\n", 0);
else printf("%d\n", 1);
```

- ☐ è sintatticamente scorretto
- ☐ è sintatticamente corretto ed il valore di output è sicuramente zero
- ☐ è sintatticamente corretto ed il valore di output è sicuramente uno
- ☐ è sintatticamente corretto ed il valore di output può essere sia zero che uno

Eventuale commento \_\_\_\_\_

## Laboratorio di Informatica

Gruppo B

**B1 – (2,5 punti)** Sia dato il seguente programma:

<pre>#include &lt;stdio.h&gt; #define D 4 float A(float *V, int k) {     int i;     float s=0.0;     for (i=0; i&lt;D; i+=k) {         V[i] = i;         s=s+V[i];     }     return s; }</pre>	<pre>void main() {     float V[] = {1.5, 2.5, 3.5, 4.5};     int i;     for (i=1; i&lt;D; i++)         V[i] = V[i-1];     printf("%f\n", A(V,2));     for (i=0; i&lt;D; i++)         printf("%f\t", V[i]);     return; }</pre>
--	--

Qual è l'uscita del programma?   =>

---

**B2 - (2,5 punti)** Dato il seguente programma, indicare quanto qui sotto richiesto:

<pre>#include &lt;stdio.h&gt; #define N 6 void main() {     int V[] = {0, 0, 0, 0, 0, 0};     int i;     for (i=0; i&lt;N; i += 3)         V[i] = V[i] + 2*i;     for (i=0; i&lt;N; i++)         if (V[i]) printf("%d\t", V[i]);         else printf("%d\t", i-V[i]);     return; }</pre>	<p><b>Valori di V dopo il 1° ciclo:</b></p> <table><tr><td></td><td></td><td></td><td></td><td></td><td></td></tr></table> <p><b>Output:</b></p> <hr/>						

## Laboratorio di Informatica

**B3 - (2,5 punti)** Nell'ipotesi che **n** sia il numero degli elementi contenuti nel vettore **vet** e che **i** sia un valore compreso tra 0 e n, dire qual è l'output del seguente frammento di programma:

```
void mystery(int *vet, int n, int i)
{
    int a;
    int j = n-1;
    while(i<j)
    {
        a = vet[i];
        vet[i++] = vet[j];
        vet[j--] = a;
    }
}
```

```
.....
int V[6] = {10, 20, 30, 40, 50, 60};
.....
mystery(V, 6, 2);
for (i=0; i<6;i++) printf("%d ", V[i]);
```

**Output:**

--	--	--	--	--	--

**B4 - (2,5 punti)** Si indichino, nel giusto ordine, i valori stampati dalle istruzioni (1) e (2) del seguente programma C, motivando la risposta data.

```
#include <stdio.h>
#define L 2
#define C 4
int MAT[L][C] = {{10,10,10,10}, {20,20,20,20}};
void main () {
    void Fun(int *v, int dim);
    int Y=10, i, j;
    Fun(&Y,L*C);
    printf("%d\n",Y);                /* (1) */
    for (i = 0; i < L; i++)
    {   for (j = 0; j < C; j++)
        printf(" %d\t",MAT[i][j]);  /* (2) */
        printf("\n");
    }
}
```

```
void Fun(int *p, int size)
{
    int i, *punt;
    (*p)++;
    punt=MAT;
    for (i=0; i<size; i=i+2) *(punt+i)=0;
    return;
}
```

**(1) :**

---

**(2) :**


*Eventuale commento* \_\_\_\_\_

## Laboratorio di Informatica

## Gruppo C

**C1 - (4 punti)** Scrivere una funzione che, dato in ingresso un array **V** di **N** numeri interi, ritorni lo stesso vettore **V** contenente gli inversi dei numeri presenti nell'array.

Gli zeri devono essere ignorati.  
Ad esempio se **V** contiene i valori

{ 10 , 2 , 0 , 5 }

la funzione dovrà ritornare

{ 0.1    0.5    0.2 }


Firma

---