

**Politecnico di Milano - Anno Accademico 2004-05 - Informatica C**  
**Prova in Itinere 27 giugno 2005 – 2<sup>a</sup> prova B**

COGNOME e NOME	
Matricola	

**RISPOSTE QUESITI TEORICI**

Indicare le risposte corrette apponendo  
una croce nella casella corrispondente.

	A	B	C	D				
1	X				2			
2			X		2			
3	X				2			
4		X			2			

5		X	X		1		-0,5	
6			X	X	1		-0,5	
7			X		1		-0,5	
8	X			X	1		-0,5	
9	X			X	1		-0,5	
<b>TOT</b>								

**PARTE 1 – RISPOSTA SINGOLA - Ogni domanda ha una sola risposta VERA.**

- Una risposta esatta fa acquisire il punteggio riportato a fianco della domanda
- Una risposta errata viene calcolata: -1
- Una risposta lasciata in bianco viene calcolata: 0

**[1] (2 pt) Qual è il ciclo corretto per trovare l'indice del primo elemento nullo (=0) in un vettore `int v[MAX]` ipotizzando `i=0` ?**

[a] `while(v[i]!=0 && i<MAX) i++;`  
 [b] `while(i<MAX) { if(v[i]==0) ++i; }`

[c] `while(i<MAX || v[i]==0) i++;`  
 [d] nessuno dei precedenti

**[2] (2 pt) Quale dei seguenti cicli calcola la somma di tutti i numeri da 1 a N, ipotizzando che `acc=0` ?**

[a] `for (i=0; i<N; i++)acc=acc + N;`  
 [b] `for (i=0; i<N; i=i+1)acc=acc + i;`

[c] `for (i=1; i<=N; i=1+i) acc= acc+i;`  
 [d] nessuna delle precedenti

**[3] (2 pt) Contrassegnare l'affermazione corretta riguardo i vettori.**

- a) È un errore inserire in una lista di inizializzazione più elementi di quanti siano presenti nel vettore
- b) Un vettore può immagazzinare contemporaneamente valori di tipo `int` e di tipo `float`
- c) L'indice di un vettore può essere del tipo di dato `double`
- d) Qualora un singolo elemento di un vettore sia passato per valore ad una funzione e sia modificato dalla stessa, nella funzione chiamante quell'elemento conterrà il valore modificato

**[4] (2 pt) Per inserire 5 numeri interi, in un array `vet` di 5 elementi, l'istruzione corretta è:**

- a) `for (i=1; i<5; i++) scanf("%d", vet );`
- b) `for (i=0; i<5; i++) scanf("%d", &vet[i] );`
- c) `for (i=0; i<5; i++) scanf("%d", &vet );`
- d) `for (i=0; i<5; i++) scanf("%d", vet[i] );`

**Politecnico di Milano - Anno Accademico 2004-05 - Informatica C**  
**Prova in Itinere 27 giugno 2005 – 2<sup>a</sup> prova B**

COGNOME e NOME	
Matricola	

**PARTE 2 – (POSSIBILI) RISPOSTE MULTIPLE - Ogni domanda può avere una o più risposte CORRETTE.**

- Ogni risposta esatta viene calcolata: +1
- Ogni risposta errata viene calcolata: -0.5
- Una risposta lasciata in bianco viene calcolata: 0

**[5] Una funzione:**

- a) Ha sempre almeno un valore di ritorno
- b) Può richiamare altre funzioni
- c) Può restituire un solo valore
- d) Ha sempre uno o più parametri

**[6] Contrassegnare la/le affermazione/i corretta/e riguardanti la funzione printf.**

- a) Serve per ricevere dati dallo standard input
- b) Quando è invocata, comincia a visualizzare sempre dall'inizio di una nuova riga
- c) Serve per scrivere in un file
- d) La sequenza di escape `\n`, quando è usata nella stringa di controllo del formato, induce il cursore a posizionarsi all'inizio della riga successiva sullo schermo

**[7]. Contrassegnare la/e affermazione/i corretta/e riguardanti i file.**

- a) La funzione `fscanf` non può essere utilizzata per leggere dati dallo standard input
- b) Il programmatore deve usare esplicitamente la `fopen` per aprire gli stream dello standard input, standard output e standard error
- c) La funzione `fprintf` può scrivere sullo standard output
- d) Per chiudere un file, un programma deve chiamare esplicitamente la funzione `fclose`

**[8]. Per definire una costante PIGRECO di valore 3.1415:**

- a) Si può utilizzare l'istruzione `const float PIGRECO = 3.1415;`
- b) Si può utilizzare l'istruzione `float const PIGRECO = 3.1415;`
- c) Si può utilizzare la direttiva `#DEFINE PIGRECO 3.1415`
- d) Si può utilizzare la direttiva `#define PIGRECO 3.1415`

**[9]. Quali di queste definizioni è corretta per definire una variabile POINTER che sia puntatore ad una funzione che accetta per argomento un array di puntatori a int e ritorna come valore un puntatore ad un float**

- a) `float *(*POINTER)(int *a[]);`
- b) `float *(*POINTER)(int (*a)[]);`
- c) `float (*POINTER)(int (*a)[]);`
- d) `float *(*POINTER)((int *)a[]);`

COGNOME e NOME

Matricola

### Parte 3 - Programmazione in C

[10]. Scrivere qui accanto un frammento di codice che stampa in ordine decrescente, uno per riga, tutti i numeri fra 800 e 200 (compresi), esclusi 222, 555 e tutti i numeri multipli di 7.

**RESTRIZIONI:** Per il ciclo non è consentito usare for. Per le decisioni non è consentito usare if.

(Suggerimento: un numero è multiplo di 7 se e solo se il resto della divisione intera per 7 di tale numero è pari a zero)

```
int i=800;
while(i>=200)
{
    switch(i)
    {
        case 222:
        case 555:
            break;
        default:
            switch(i%7)
            {
                case 0: /* multiplo di 7! */
                    break;
                default: /* non mult. di 7! */
                    printf("%d\n",i); break;
            }
            i--;
    }
}
```

\* \* \*

Supponendo che sia stato dichiarato il tipo

```
struct nodo
{
    int val;
    struct nodo * next;
};
```

e che tale tipo sia stato usato per costruire una lista singola (attualmente contenente 10 elementi), il cui primo elemento è raggiungibile usando un puntatore **head**,

[11] scrivere **una singola istruzione** che assegna alla variabile intera **valore** (che si suppone già dichiarata) il valore del campo **val** del secondo elemento della lista.

**valore = head->next->val;**

### DOMANDA SU TEMI DI LABORATORIO

Il seguente programma contiene alcuni errori. **Riscriverlo corretto** nello spazio accanto, evidenziando quali errori sarebbero stati rilevati dal compilatore (**indicare COMP**) e quali invece si sarebbero manifestati solo in esecuzione. (**indicare ESEC**)

```
#include <stdio.h>
int main
{
    if(i=1)
        printf('i vale 1');
    switch i
    case "1";
        printf('i vale 1');
    case "2";
        printf('i vale 2');
}
```

```
<> #include <stdio.h> COMP
() int main() COMP
{
    dich. i
    int i; COMP
    if(i==1) ESEC
        printf("i vale 1"); COMP
    switch (i) COMP
    { COMP
        case 1: COMP
            printf("i vale 1"); COMP
            break; ESEC
        case 2:
            printf("i vale 2"); COMP
            break; ESEC
    }
}
```

**Politecnico di Milano - Anno Accademico 2004-05 - Informatica C**  
**Prova in Itinere 27 giugno 2005 – 2^ prova B**

COGNOME e NOME	
Matricola	

Supponendo che sia stato dichiarato un array **collezione** di 100 strutture **disco** aventi la seguente definizione:

```
struct disco
{
    int durata_minuti;
    char titolo[200];
    char tipo;
};
```

[12] scrivere un frammento di codice che acquisisce da tastiera la durata in minuti del disco rappresentato dal terzo elemento dell'array, visualizzando prima un opportuno messaggio di richiesta.

```
printf("Inserisci la durata in minuti del disco:");
scanf("%d",&(collezione[2].durata_minuti));
```

[13] Supponendo già dichiarata la variabile **char buffer[100]**, scrivere un frammento di codice che apre il file di testo "prova.txt" in lettura (verificando se l'apertura è riuscita o fallita) e poi ne legge per righe l'intero contenuto (usando **buffer** come variabile di appoggio) e visualizza, di ogni riga letta, il solo primo carattere (andando a capo ogni volta). Per esempio:

Contenuto del file prova.txt:

pippo  
ciao  
Finito

Risultato che deve apparire su schermo in questo caso:

p  
c  
F

```
char buffer[100];
FILE * inputfile=fopen("prova.txt","r");
if(!inputfile){printf("Impossibile aprire il file!"); return -1;}
while(!feof(inputfile))
{
    fgets(buffer,99,inputfile);
    printf("%c\n",buffer[0]);
}
fclose(inputfile);
```