

	Politecnico di Milano Facoltà di Ingegneria Industriale INFORMATICA B Appello 11 Febbraio 2013		COGNOME E NOME				
	RIGA	COLONNA	MATRICOLA				
			Spazio riservato ai docenti <table border="1" data-bbox="1249 526 1517 589"> <tr> <td></td> <td></td> <td></td> <td></td> </tr> </table>				

- Il presente plico contiene 4 esercizi e **deve essere debitamente compilato con cognome e nome, numero di matricola.**
- Il tempo a disposizione è di 1 ora e 30 minuti.
- Non separate questi fogli. Scrivete la soluzione solo sui fogli distribuiti, utilizzando il retro delle pagine in caso di necessità. Cancellate le parti di brutta (o ripudiate) con un tratto di penna.
- Ogni parte non cancellata a penna sarà considerata parte integrante della soluzione.
- È possibile scrivere a matita (e non occorre ricalcare al momento della consegna!).
- **È vietato utilizzare calcolatrici, telefoni o pc.** Chi tenti di farlo vedrà annullata la sua prova.
- È ammessa la consultazione di libri e appunti, purché con pacata discrezione e senza disturbare.
- **Qualsiasi tentativo di comunicare con altri studenti comporta l'espulsione dall'aula.**
- È possibile ritirarsi senza penalità.
- **Non è possibile lasciare l'aula conservando il tema della prova in corso.**
- L'esame orale è parte integrante dell'esame e deve essere realizzato almeno sufficientemente per il superamento dell'esame complessivo.
- Per l'ammissione all'esame orale è necessario aver almeno impostato sufficientemente entrambi gli esercizi 1 e 2.

Esercizio 1 (10 punti)

Una fattoria gestisce una stalla con 10 mucche, ciascuna identificata da un codice alfanumerico di 8 caratteri.

Il gestore della fattoria vuole realizzare un programma in C che consenta la contabilizzazione del latte prodotto in un giorno dalle 10 mucche.

In questo contesto, si risponda ai seguenti punti:

1. Si definisca in C il tipo **ProduzioneGiornaliera** che rappresenti la quantità di latte prodotta da ognuna delle 10 mucche della stalla. La produzione giornaliera di ogni mucca deve essere associata al suo codice identificativo. Si definiscano quindi ulteriori tipi se necessario. La quantità di latte prodotta da ciascuna mucca è espressa in litri.
2. Si definisca la variabile **produzione**, utilizzando in modo appropriato il tipo **ProduzioneGiornaliera** definito al punto 1, per la memorizzazione della produzione giornaliera delle 10 mucche della stalla. Si supponga inoltre che la variabile **produzione** sia stata opportunamente inizializzata con la produzione giornaliera delle 10 mucche della stalla.
Si scriva un frammento di programma C che stampi a video il codice identificativo e la quantità di latte prodotta da ciascuna mucca della stalla. Si stampi inoltre la quantità totale di latte prodotto nella stalla.

SOLUZIONE

```
typedef char stringa[8];

typedef struct {
    stringa codiceMucca;
    float litri;
} ProduzioneMucca;

typedef ProduzioneMucca ProduzioneGiornaliera[10];

ProduzioneGiornaliera produzione;

float tot = 0;

for (i = 0; i < 10 ; i++)
{
    tot += produzione[i].litri;
    printf("\nla mucca %s ha prodotto %f litri", produzione[i].codiceMucca, produzione[i].litri);
}

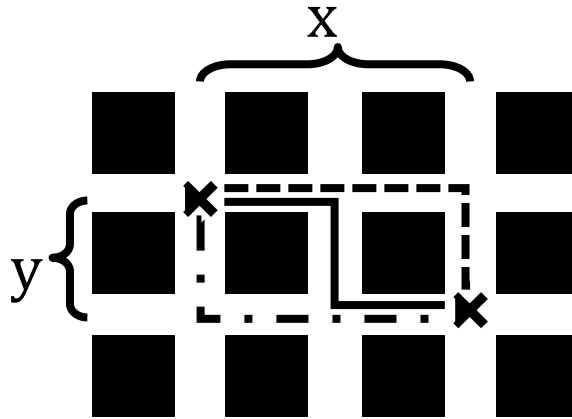
printf("\n\tProduzione totale %f", tot);
```

Esercizio 2 (10 punti)

Le strade della città di Grigliopoli sono organizzate come una griglia (alcune strade attraversano la città da est a ovest e altre da nord a sud). Dati due incroci che distano X isolati lungo l'asse est-ovest della città e Y isolati lungo l'asse nord-sud, si vuole calcolare il numero di percorsi a distanza minima che collegano i due incroci. Implementare una funzione ricorsiva *calcola* in Matlab/Octave che ricevuti X e Y in ingresso restituisce il numero di percorsi trovati in uscita.

Suggerimento: Quando $X = 0$ o $Y = 0$, c'è soltanto un cammino a distanza minima che collega i due incroci. Altrimenti, esiste più di un cammino minimo dal momento che è possibile sia avvicinarsi alla destinazione lungo l'asse est-ovest (riducendo quindi la distanza X) oppure avvicinarsi lungo l'asse nord-sud (riducendo la distanza Y)

Esempio: Nell'esempio qui sotto, vengono mostrati i 3 percorsi a distanza minima che collegano due incroci a distanza $x = 2$ e $y = 1$



Soluzione

```
function p = calcola(x,y)
    if (x == 0 || y == 0)
        p = 1;
    else
        p = calcola(x-1,y) + calcola(x,y-1);
    end
```

Esercizio 3 (6 punti)

Si consideri il seguente programma in linguaggio C:

```
#define N -17

int main() {
    int a = N/10;
    float b = N/10.0;
    double c = N/10.0;
    double d = b;
}
```

Sapendo che i tipi delle variabili utilizzate sono codificati in binario in questo modo:

- **int**: complemento a due a 32 bit
- **float**: virgola mobile a precisione singola secondo lo standard IEEE 754-1985 (1 bit per il segno, 23 bit per la mantissa, 8 per l'esponente ($K = 127$))
- **double**: virgola mobile a precisione doppia secondo lo standard IEEE 754-1985 (1 bit per il segno, 52 bit per la mantissa, 11 per l'esponente ($K = 1023$))

Si risponda alle seguenti domande:

1. Qual è il valore in decimale e in binario della variabile **a** alla fine dell'esecuzione dell'ultima istruzione? Giustificare la risposta mostrando i calcoli effettuati.
2. Qual è il valore in decimale e in binario della variabile **b** alla fine dell'esecuzione dell'ultima istruzione? Giustificare la risposta mostrando i calcoli effettuati.
3. Considerando il risultato ottenuto nel punto 2, alla fine dell'esecuzione dell'ultima istruzione le variabili **c** e **d** contengono lo stesso valore binario? La risposta cambierebbe se la costante **N** fosse definita pari a 10 anziché a -17? Non è necessario ricavare il valore in binario di **c** e **d**, ma la risposta deve essere adeguatamente motivata.

Soluzione

1. Valore di a in decimale: $a = -17/10 = -1$. Il risultato si tronca perché è intero.
Rappresentazione binaria di 1 come numero naturale a 32 bit:
00000000000000000000000000000001, in complemento a due: -1 è
11111111111111111111111111111110 + 1 = 11111111111111111111111111111111
Quindi il valore di a in binario è: 11111111111111111111111111111111
2. Valore di a in decimale: $b = -17/10 = -1.7$
S = 1 (segno negativo)
parte intera del numero: 1
determino la parte frazionaria del numero:
 $0.7 \times 2 \rightarrow 1$
 $1.4 \times 2 \rightarrow 0 *$
 $0.8 \times 2 \rightarrow 1 *$
 $1.6 \times 2 \rightarrow 1 *$
 $1.2 \times 2 \rightarrow 0 *$
0.4 ... (il periodo e' 0110)
parte frazionaria del numero: .10110
 $M = 1.10110 \times 2^0$ (la mantissa è già normalizzata) = (sottintendendo 1. e troncando il numero al 23esimo bit della precisione singola) = 10110011001100110011001
 $E = 127 + 0 = 127_{10} = 01111111_2$. Quindi il valore di b in binario è:
10111111110110011001100110011001
3. Dall'analisi dei calcoli effettuati nel punto 2 emerge che la rappresentazione del valore -1.7 in binario è periodica. Questo significa che tale valore sarà diverso a seconda che venga calcolato in precisione singola o doppia. Entrambe le variabili **c** e **d** sono double e consentono quindi di rappresentare numeri in precisione doppia. Alla variabile **d** però viene assegnato il valore di **b** che è già troncato in precisione singola. Per questo motivo, **c** e **d** avranno valori diversi. Per **N = 10** **c** e **d** hanno lo stesso valore in binario perché il valore 1 (10/10) nella codifica binaria in virgola mobile non è periodico e non viene approssimato né in precisione singola né in precisione doppia.