	Politecnico di Milano Facoltà di Ingegneria Industriale INFORMATICA B Prova in itinere del 28 gennaio 2013		COGNOME E NOME
	RIGA	COLONNA	MATRICOLA
			<i>Spazio riservato ai docenti</i>
			<input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/>

- Il presente plico contiene 3 esercizi e **deve essere debitamente compilato con cognome e nome, numero di matricola.**
- Il tempo a disposizione è di 1 ora e 30.
- 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.**
- **Saranno valutate meno le soluzioni che NON sfruttano i vantaggi sintattici del linguaggio Matlab/Octave nella gestione degli array.**

Esercizio 1 (7 punti)

Il Sig. Rossi ha predisposto investimenti utilizzando svariati strumenti finanziari (azioni, obbligazioni, titoli di stato, etc..) e ha annotato:

- Il capitale iniziale di ogni investimento (a Gennaio 2012); tale capitale varia da investimento a investimento.
- Il valore corrente di ogni investimento (a Gennaio 2013).

Il Sig. Rossi a Gennaio 2012 ha preventivato di ottenere una rendita minima dai suoi investimenti. La rendita minima è uguale per tutti gli investimenti ed è espressa come una percentuale; è quindi un numero nell'intervallo $[0,1]$ (ad esempio $+3\% = 0.03$).

Si assuma che il capitale iniziale e il valore corrente degli investimenti vengano ordinatamente salvati in due vettori di uguale lunghezza, e che quindi l'indice di questi vettori identifichi univocamente l'investimento.

Sviluppare una funzione Matlab/Octave chiamata **controllaRendita** che, a partire dalla rendita minima e dai vettori che contengono i dati sul capitale iniziale e valore corrente di ogni investimento, restituisce gli indici degli investimenti che sono cresciuti di una percentuale maggiore o uguale alla rendita minima prevista.

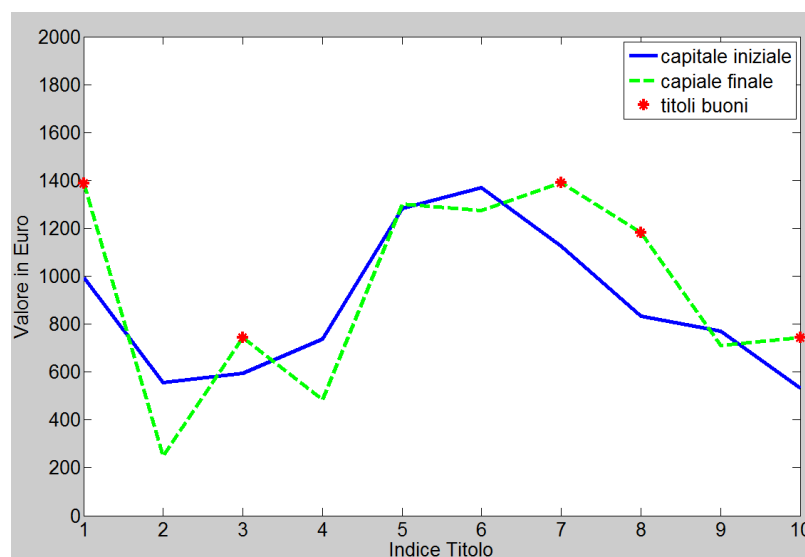
A) Si definisca l'intestazione della funzione **controllaRendita** spiegando il significato dei suoi parametri in ingresso e valori di ritorno.

B) Si scriva il codice della funzione **controllaRendita**.

C) Si costruisca uno script che:

- carica dal file ascii *capitaleIniziale.txt* gli importi del capitale iniziale per ogni investimento a Gennaio 2012 e dal file ascii *valoreCorrente.txt* i valori correnti degli investimenti (ogni file contiene un numero per ogni riga)
- chiede all'utente di inserire la rendita minima
- richiama la funzione **controllaRendita** sviluppata al punto precedente
- visualizza un grafico che contiene sull'asse delle x i valori interi che identificano gli investimenti e sull'asse delle y i corrispondenti valori in euro degli investimenti. Il grafico riporta:
 - una linea blu per indicare il capitale iniziale di ogni investimento
 - una linea verde per indicare il valore corrente di ogni investimento
 - un punto rosso in corrispondenza agli indici e ai valori dell'investimento corrente per tutti gli investimenti che sono cresciuti almeno quanto la rendita minima.

Di seguito viene riportato un esempio del grafico richiesto



- genera e visualizza a video una tabella contenete i soli titoli buoni; tale tabella deve contenere nella prima colonna la posizione (riga) in cui ogni titolo buono si trova nei file *capitaleIniziale.txt* e *valoreCorrente.txt*, nella seconda colonna il valore corrente di ogni titolo buono e nella terza colonna la rendita (o perdita in caso fosse negativa) corrente. [Per calcolare quest'ultimo valore si usi la formula: $rendita(ii) = (valoreCorrente(ii) / capitaleIniziale(ii)) - 1$]

Soluzione

Funzione

```
function titoliBuoni = controllaRendita(capitaleIniziale, valoreCorrente, renditaMinima)
% function titoliBuoni = controllaRendita(capitaleIniziale, valoreCorrente, renditaMinima)
%
% input:
% capitaleIniziale: vettore contenente il valore degli investimenti a Gennaio 2012
% valoreCorrente: vettore contenente il valore attuale degli investimenti
% renditaMinima: scalare contenente il valore della rendita minima che ci si aspetta da ogni investimento
%
% output:
% titoliBuoni: vettore contenente gli indici di capitaleIniziale che corrispondono a investimenti che hanno reso
% almeno quanto renditaMinima
%
titoliBuoni = find(valoreCorrente ./ capitaleIniziale >= 1 + renditaMinima);
```

Script

```
capitaleIniziale = load('capitaleIniziale.txt');
valoreCorrente = load('valoreCorrente.txt');
renditaMinima = input('Inserire il valore della rendita minima (in intervallo [0,1]): ');
titoliBuoni = controllaRendita(capitaleIniziale, valoreCorrente, renditaMinima);
x = [1 : length(capitaleIniziale)];
figure()
plot(x, capitaleIniziale, 'b');
hold on
plot(x, valoreCorrente, 'r');
plot(titoliBuoni, valoreCorrente(titoliBuoni), 'go');
legend('capitale iniziale', 'valore corrente', ['titoli con rendita >= ', num2str(renditaMinima)]);
hold off
linee = 1 : length(capitaleIniziale);
rendita = valoreCorrente ./ capitaleIniziale - 1;
tabella = [linee', valoreCorrente rendita];
tabella(titoliBuoni, :) = [];
disp(tabella);
```

Esercizio 2 (6 punti)

Scrivere la funzione ricorsiva **contaNonMultipli** che prende in ingresso due interi **num** e **n** (entrambi strettamente positivi). La funzione **contaNonMultipli** conta quante cifre è necessario rimuovere a **num**, a partire da destra, prima di incontrare un multiplo di **n**. Appena la funzione trova un multiplo di **n** restituisce tramite il parametro di uscita quante cifre è stato necessario rimuovere in coda a **num** prima di ottenere tale numero divisibile per **n**. Se **num** è nullo, la funzione restituisce 0; se nessun multiplo di **n** è stato trovato, la funzione restituisce il numero di cifre che sono state rimosse (ovvero il numero di cifre che compongono **num**).

Esempi

- `contaNonMultipli(12333, 2)` restituisce 3 perché 12333, 1233, 123 non sono multipli di 2, mentre 12 lo è.
- `contaNonMultipli(12300, 2)` restituisce 0 perché 12300 è multiplo di 2.
- `contaNonMultipli(0, 3)` restituisce 0 perché `num` è nullo.
- `contaNonMultipli(123, 5)` restituisce 3 perché 123, 12 e 1 non sono multipli di 5.

Suggerimento: un numero intero positivo di n cifre, c_n, \dots, c_1 può essere rappresentato in termini di potenze di 10 come segue: $c_n \cdot 10^{n-1} + c_{n-1} \cdot 10^{n-2} + \dots + c_2 \cdot 10 + c_1$. Per esempio $1456 = 1 \cdot 10^3 + 4 \cdot 10^2 + 5 \cdot 10 + 6$. Porzioni del numero possono essere ricavate dividendolo per un'opportuna potenza di 10. Per esempio, la divisione intera tra 1456 e 10 dà come risultato 145.

Soluzione

```
function ris = contaNonMultipli(num , divisore)
```

```
if (num == 0)
```

```
    ris = 0;
```

```
elseif (mod(num , divisore) == 0)
```

```
    ris = 0;
```

```
else
```

```
    ris = 1 + contaNonMultipli(floor(num/10), divisore);
```

```
end
```

Esercizio 3 (4 punti)

Un sistema dotato solamente di memoria centrale ha un tempo di accesso ai dati pari a 150 ns. Per poter migliorare il tempo di accesso si decide di aggiungere una memoria cache dalle seguenti caratteristiche:

- Miss Rate = 20 %
- Miss Penalty = 500 ns
- Hit Time = 100 ns

Rispondere alle seguenti domande (giustificando i risultati ottenuti con gli opportuni calcoli):

- Calcolare il tempo medio di accesso ai dati.
- Qual è il valore massimo che deve avere l'Hit Time affinché la cache offra un vantaggio in termini di prestazioni (cioè il tempo medio di accesso con la cache sia inferiore a 150 ns)?
- Il progettista suggerisce che è possibile incrementare l'Hit Rate con i seguenti accorgimenti:
 - quando vengono caricati dei dati in cache, se la memoria cache è piena è opportuno che i nuovi dati sostituiscano quelli più recentemente caricati in cache.
 - in caso di miss è opportuno portare in cache non solo il dato richiesto, ma anche un (piccolo) blocco di dati che si trovano nelle posizioni immediatamente successive al dato richiesto.

Commentare gli accorgimenti (i) e (ii) proposti dal progettista e in particolare valutarne l'efficacia in riferimento ai principi utilizzati per gestire la memoria nei sistemi operativi.

Soluzione

a) $TM = (1 - 0.2) * 100 + 0.2 * 500 = 80 + 100 = 180 \text{ ns}$

b) $(1 - 0.2) * x + 0.2 * 500 \leq 150$
 $0.8 * x \leq 50 \rightarrow x \leq 50 * (5 / 4) = 62,5 \text{ ns}$ è l'Hit time massimo accettabile affinché la cache sia conveniente

c) L'accorgimento (i) è in disaccordo con il principio di località temporale e quindi potrebbe essere dannoso. L'accorgimento (ii) è in accordo con il principio di località spaziale e può quindi portare a miglioramenti dell'hit rate.