

Esame di Laboratorio di Fisica Computazionale

19 settembre 2017, ore 9.30

1 Mathematica

1. Si definisca una funzione `myFactorial` che prende in input un intero n positivo e restituisce il fattoriale $n!$.
(*Si possono usare i comandi nativi di Mathematica con l'esclusione di `Factorial` ovvero di `!`.*)

2. Si definisca una funzione `myBinomial` che prende in input due interi n e k con $n > k$ e restituisce il coefficiente binomiale $\binom{n}{k}$.
(*Si possono usare i comandi nativi di Mathematica con l'esclusione di `Binomial`.*)

3. Si definisca una funzione `myDet3` che prende in input tre vettori colonna \vec{l}_1 , \vec{l}_2 e \vec{l}_3 , ovvero tre liste costituite ciascuna da tre elementi, e restituisce in output il determinante della matrice $M \equiv (\vec{l}_1, \vec{l}_2, \vec{l}_3)$.
(*Si possono usare i comandi nativi di Mathematica con l'esclusione di `Det`.*)

Suggerimento: si utilizzi il comando `Signature`.

4. Si definisca una funzione dei parametri a e b come la soluzione dell'equazione differenziale seguente:

$$r'(\theta) = br(\theta), \quad r(0) = a \quad (1)$$

Si disegni la curva parametrica espressa in coordinate polari dalla coppia $(r(\theta), \theta)$ al variare del parametro $\theta \in [0, 6\pi]$ avendo posto $a = b = 1$.

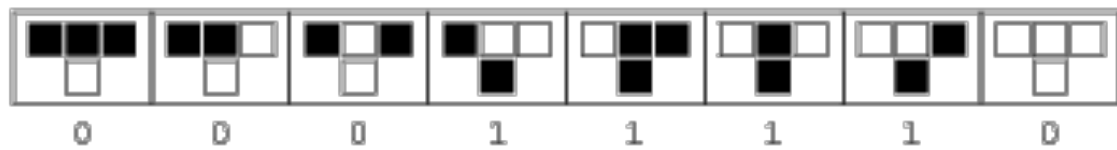
5. Si definisca una funzione dei parametri a e b come la soluzione dell'integrale

$$p(a, b) \equiv \int_a^b dt \frac{1}{\sqrt{(t-a)(b-t)}} \quad (2)$$

Si disegni in funzione con $a \in [-5, 5]$ e con $b \in [-5, 5]$ la superficie $p(a, b)$.

6. Automa cellulare

- Si scriva una funzione che inizializza una lista contenente $2n + 1$ elementi, tutti zero con l'eccezione dell'elemento $n + 1$, che deve essere posto pari a 1.
(*Si consideri l'utilizzo combinato di `ReplacePart` e di `Table`.*)
- Si scriva una lista di regole di sostituzione che realizzi le trasformazioni rappresentate graficamente in figura.
(*Si consideri per la singola sostituzione una regola del tipo $\{a, b, c\} \rightarrow d$*)



- La trasformazione di una lista esistente in una nuova, secondo il meccanismo dell'automa cellulare, richiede di applicare a ciascuna terna di elementi consecutivi della lista il set di regole di sostituzione del punto precedente. Per semplicità, il primo e l'ultimo elemento della lista possono essere ricopiati da una riga all'altra, mentre si devono calcolare tutti gli elementi nuovi dal secondo al penultimo.

*Si utilizzino i comandi **Table** per generare i nuovi elementi, dal secondo al penultimo, e poi **Prepend** e **Append** per ricopiare il primo e l'ultimo.*

2 C++: Esercizio

Si risolvano i due esercizi proposti, preferibilmente in due file singoli separati. La sufficienza è raggiunta risolvendo correttamente i primi tre punti del primo esercizio.

Esercizio 1

1. Si scriva una classe **Vector** che descriverà vettori in due dimensioni. Si pongano come membri privati due variabili **double**, corrispondenti alle componenti cartesiane del vettore (ad esempio x e y). Si scriva un opportuno costruttore che richieda come parametri le due coordinate, con valori di default nulli. Si implementino inoltre due *access function* per leggere le due variabili private.
2. Si scriva il costruttore di copie (con la sintassi della lista di inizializzazione). Inoltre si scriva l'operatore “+=” che esegua la somma con un altro vettore, e una funzione **theta** che restituisca l'angolo che il vettore forma con l'asse x . [Nota: la funzione **atan** ha valori in $(-\pi/2, \pi/2)$.]
3. Si scriva la classe **Versor**, derivata di **Vector**, che descriverà vettori di norma unitaria. Si scrivano due costruttori: uno che prenda come parametro l'angolo rispetto all'asse x , e l'altro che prenda un **Vector** e ne costruisca il versore corrispondente. Bonus se entrambi i costruttori hanno corpo vuoto, cioè {}.
4. Si scriva l'*overriding* dell'operatore “+=”, in modo che sommare un vettore v ad un versore u generi un versore diretto come $u + v$. Si vuole che questa funzione abbia comportamento polimorfico tra **Vector** e **Versor**.
5. Si verifichi il comportamento del codice con un **main** che esegua le seguenti istruzioni:
 - istanziare due vettori v e w , inizializzandoli a $(1, 1)$ e $(2, 2)$ rispettivamente;
 - istanziare un versore u , inizializzandolo con v ;
 - eseguire $u += w$;
 - stampare su **std::cout** le componenti di u .
6. Per verificare il comportamento polimorfico dell'operatore “+=”, si ripetano gli stessi passi del punto precedente, utilizzando però tre puntatori (a v , w e u), e si controlli che il risultato sia lo stesso.

Esercizio 2

Si consideri il tipo **std::vector<int>** (per comodità lo chiameremo “Sequenza”; se si vuole si può definire un alias con **typedef**, ma non è necessario).

1. Si scriva una funzione `vsort`, che ordinerà in senso crescente gli elementi di un `std::vector` (passato come argomento) contenente oggetti di tipo Sequenza. La relazione di ordinamento è la seguente: una Sequenza r è minore di s se la somma degli elementi di r è minore della somma degli elementi di s . Si usi l'algoritmo `std::sort(it1, it2, pred)`. Per fare questo sarà necessario definire un opportuno predicato `pred`, che confronti due oggetti di tipo Sequenza. [Si calcoli la somma degli elementi con l'algoritmo `std::accumulate`].
2. Nel `main`, si istanzi un vettore di sequenze, e vi si inseriscano le due sequenze $\{3,5,7\}$ e $\{1,1,2,3,5\}$. Lo si ordini e si stampi il terzo elemento della prima sequenza, controllando che sia uguale a 2.
3. Si scriva una versione *generica* della funzione `vsort`, che possa funzionare anche con vettori di sequenze di oggetti di tipo diverso da `int`. Si controlli, commentando il codice non generico scritto al punto 1, che il programma funzioni correttamente.
4. Si renda generico anche il predicato `pred`, con la stessa parametrizzazione descritta al punto precedente.

3 Shell scripting

1. Si scriva uno script che scambia le due colonne del file `prova.txt` e salva il risultato nel file `scambiate.txt`.
2. Si scriva uno script che calcola la media dei valori della terza colonna del file `voti.txt`
3. Si scriva uno script che seleziona, tra tutti i files presenti nella directory `/home/vicini/corso1617/`, quelli creati nel mese di aprile 2017 e li scrive nel file `aprile.txt`.