

RELAZIONE-PROGETTO INGEGNERIA DEGLI ALGORITMI

Michele Terenzi

Matricola:0254291

Scelte implementative :

L'algoritmo quicksort scelto che utilizzerà i tre algoritmi di selezione per calcolare il pivot della lista (Selezione randomizzata, Selezione deterministica e sampleMedianSelect) sarà quello di tipo ricorsivo.

Nel calcolo del sottoinsieme V della lista l è stata utilizzata una lunghezza m generata randomicamente (in questo caso da 0 a 5 elementi).

Per il calcolo del mediano del sottoinsieme V nella funzione sampleMedianSelect() sono state utilizzati tre algoritmi preimplementati insertionSortDown(), trivalSelect() e quickSelectRand() in base alla lunghezza della lista presa in considerazione.

Per stampare il grafico dei risultati sperimentali dei nuovi algoritmi di ordinamento e confrontarli con gli altri sono state utilizzate le librerie matplotlib, numpy e pandas.

La funzione read_csv messa a disposizione dalla libreria pandas permette di leggere e caricare in un dataframe (una matrice) ogni file contenente i risultati del testing effettuato su un determinato algoritmo di ordinamento, la libreria matplotlib viene utilizzata per stampare e salvare in un file immagine il grafico e la libreria numpy permette di generare una variazione randomica del colore per facilitare la lettura del grafico.

Risultati sperimentali:

Grafici:

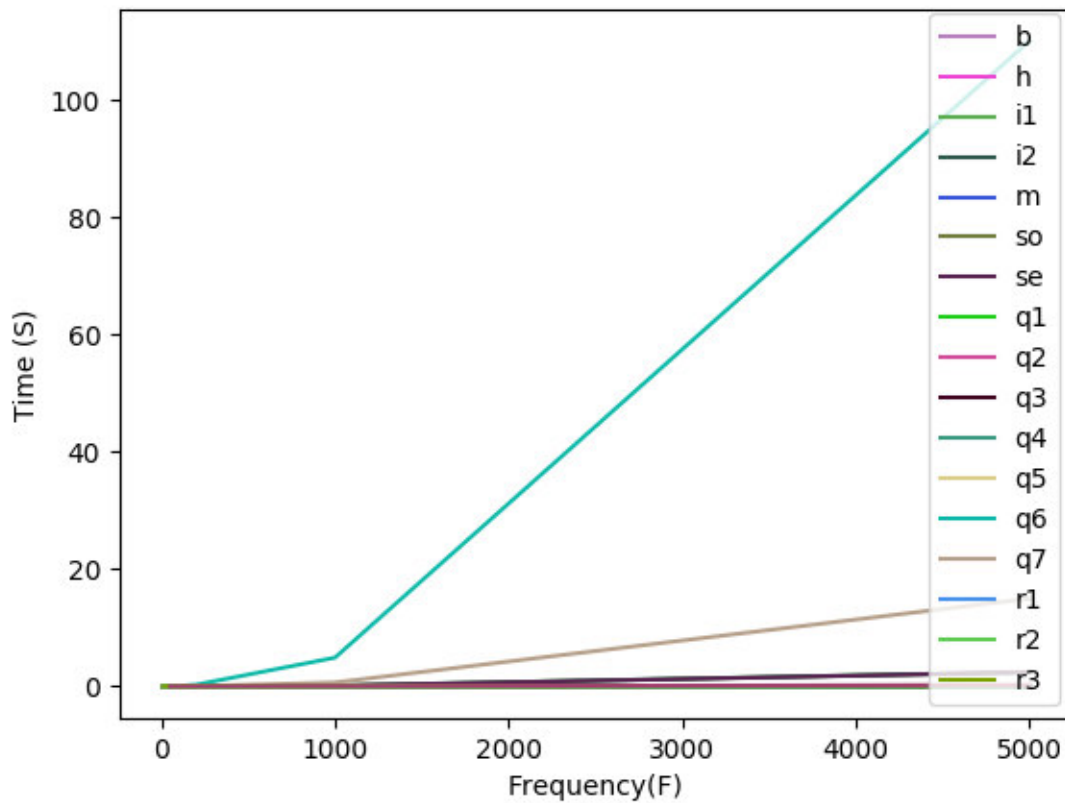
Le sigle delle etichette riportate nei grafici corrispondono rispettivamente ai seguenti algoritmi di ordinamento:

- b = Bubble Sort
- h = Heap Sort
- i1 = Insertion Sort Up
- i2 = Insertion Sort Down
- m = Merge Sort
- so = Metodo sort() di python
- se = Selection Sort
- q1 = Quick Sort Iterativo Deterministico
- q2 = Quick Sort Iterativo Non Deterministico
- q3 = Quick Sort Ricorsivo Deterministico
- q4 = Quick Sort Ricorsivo Non Deterministico
- q5 = Quick Sort Ricorsivo con sampleMedianSelect
- q6 = Quick Sort Ricorsivo con selezione deterministica
- q7 = Quick Sort Ricorsivo con selezione non deterministica
- r1 = Radix Sort con base 400
- r2 = Radix Sort con base 100

r3 = Radix Sort con base 10

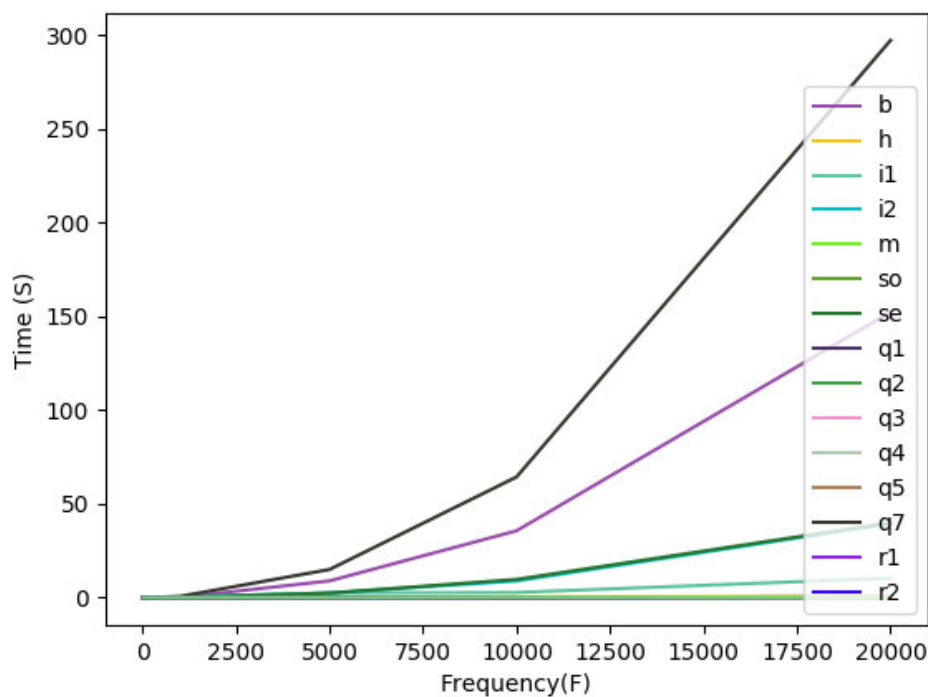
Andamento algoritmi con quickSortDetSelect() e slow algorithms

Andamento algoritmi



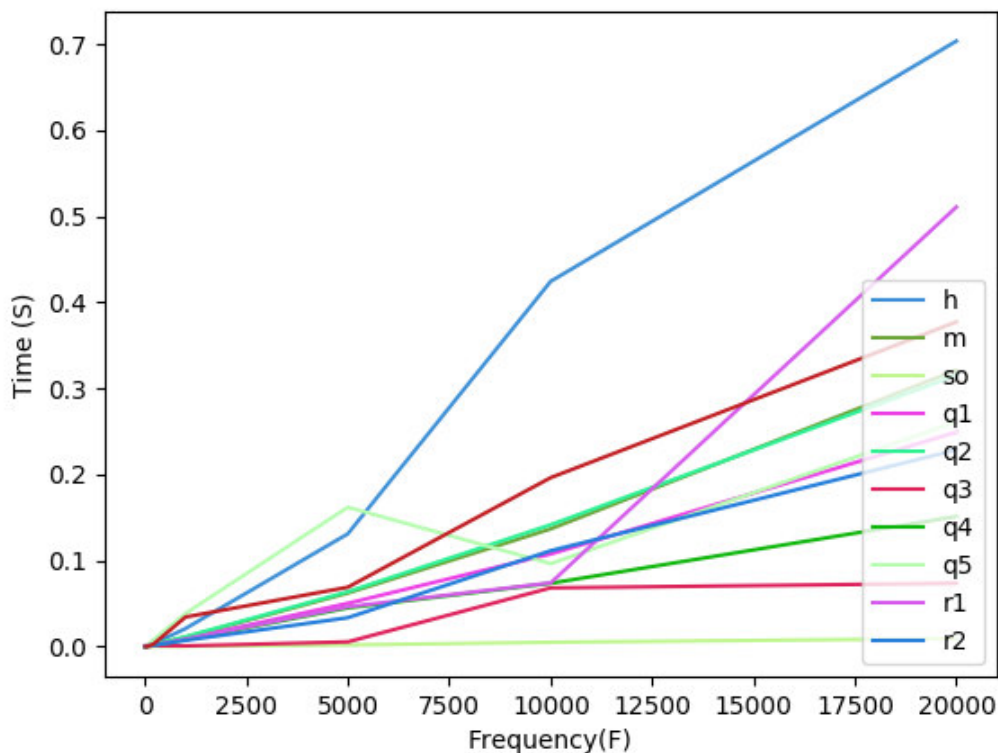
Andamento algoritmi con slow algorithms inclusi e quickSortDetSelect() esclusi

Andamento algoritmi



Andamento algoritmi con slow algorithms, quickSortRandSelect e quickSortDetSelect() esclusi

Andamento algoritmi



Commento:

Dai risultati ottenuti si evince che nella selezione del pivot dell'algoritmo quicksort randomizzato, l'algoritmo di selezione deterministico (quickSelectDet()) è il più lento al crescere della dimensione dell'input e rende il quick sort randomico il più lento fra gli algoritmi di ordinamento presi in considerazione.

Inoltre la versione del quicksort randomico che utilizza invece l'algoritmo di selezione risulta la seconda più lenta al crescere della dimensione dell'input.

La selezione randomizzata sappiamo infatti che è in genere più veloce di quella deterministica nella pratica.

Il quickSort risulta invece tra i migliori nel caso si utilizzi l'algoritmo sampleMedianSelect, poichè a differenza degli altri 2, il mediano verrà calcolato su un sottoinsieme V di lunghezza m e non maggiore della lunghezza della lista scelta dall'utente, di conseguenza scegliendo un valore pari o intorno a m costante al crescere dell'input il tempo di esecuzione sarà di gran lunga inferiore rispetto agli altri algoritmi di selezione utilizzati.

TABULATI**BUBBLE SORT**

Frequency	Time(s)
10	2,36034E-05
20	0,00011611
50	0,000778913
100	0,003214121
200	3,74317E-05
1000	0,076856375
5000	8,908909082
10000	35,62100887
20000	152,201405

INSERCTION SORT UP

Frequency	Time(s)
10	2,72E-05
20	5,89E-05
50	0,000474691
100	0,000833273
200	0,003770351
1000	0,111871481
5000	2,438637495
10000	2,711280346
20000	10,36952472

SELECTION SORT

Frequency	Time(s)
10	2,14577E-05
20	5,57899E-05
50	0,000273228
100	0,001211882
200	0,004171133
1000	0,095693111
5000	2,388968706
10000	9,689725876
20000	39,8945539

INSERCTION SORT DOWN

Frequency	Time(s)
10	2,21729E-05
20	5,6982E-05
50	0,000265121
100	0,000870705
200	0,003770351
1000	0,111871481
5000	2,438637495
10000	8,86690855
20000	39,44560766

HEAP SORT

Frequency	Time(s)
10	0,000106573
20	0,000210285
50	0,000605583
100	0,001384974
200	0,003167868
1000	0,020963907
5000	0,131028652
10000	0,424309492
20000	0,703502417

QUICK SORT ITERATIVO DET

Frequency	Time(s)
10	9,417533875E-05
20	0,000150680542
50	0,0003871917725
100	0,0008525848389
200	0,0015463829041
1000	0,0089819431305
5000	0,0502464771271
10000	0,1075091362
20000	0,2491216659546

QUICK SORT

Frequency	
10	
20	
50	
100	
200	
1000	
5000	
10000	
20000	

MERGE SORT

Frequency	Time(s)
10	6,19888E-05
20	0,000127792
50	0,000417948
100	0,000781775
200	0,001844406
1000	0,010755539
5000	0,062182903
10000	0,136730433
20000	0,321204424

QUICK SORT ITERATIVO NON DET

Frequency	Time(s)
10	0,000105381012
20	0,0001857280731
50	0,0004794597626
100	0,0009341239929
200	0,0022807121277
1000	0,011492729187
5000	0,0637452602386
10000	0,1415159702301
20000	0,3151476383209

QUICK SORT

Frequency	
10	
20	
50	
100	
200	
1000	
5000	
10000	
20000	

RICORSIVO DET

Time(s)
 3,24249267578125E-05
 9,82284545898438E-05
 0,000167608261108
 0,000370264053345
 0,000993013381958
 0,000781297683716
 0,00547981262207
 0,068183183670044
 0,073525428771973

QUICK SORT RICORSIVO CON SAMPLEMEDIAN SELECT

Frequency	Time(s)
10	0,000455617904663
20	0,00053596496582
50	0,001382827758789
100	0,003126621246338
200	0,007480382919312
1000	0,038531541824341
5000	0,161953687667847
10000	0,096261024475098
20000	0,261213302612305

RICORSIVO NON. DET

Time(s)
 5,26905059814453E-05
 0,000103235244751
 0,000292778015137
 0,000621318817139
 0,001471996307373
 0,008015394210815
 0,044942378997803
 0,073525428771973
 0,151138305664063

QUICK SORT CON SELEZIONE DETERMINISITCA

Frequency	Time(s)
10	0,000937938690186
20	0,002716779708862
50	0,012052774429321
100	0,055971384048462
200	0,263408184051514
1000	4,89386582374573
5000	110,196991443634

QUICK SORT CON SELEZIONE NON DETERMINISTICA

Frequency	Time(s)	
10		0,000483274459839
20		0,000803709030151
50		0,003968715667725
100		0,010041952133179
200		0,032638549804688
1000		0,653159618377685
5000		14,9828729629517
10000		64,3312878608704
20000		297,118931531906

RADIX CON BASE 400

Frequency	Time(s)	
10		0,000932931900024
20		0,000168085098267
50		0,000942230224609
100		0,001186847686768
200		0,001535415649414
1000		0,00862193107605
5000		0,04625129699707
10000		0,073981523513794
20000		0,510722637176514

RADIX CON BASE 100

Frequency	Time(s)
10	0,000196934
20	0,000223398
50	0,000316381
100	0,001127958
200	0,001560688
1000	0,00716877
5000	0,033548832
10000	0,111360788
20000	0,22846365

RADIX CON BASE 10

Frequency	Time(s)
10	0,000109196
20	0,000169277
50	0,000350475
100	0,000993013
200	0,002568245
1000	0,034345865
5000	0,068961143
10000	0,196249962
20000	0,377231598