Fonaments de Programació II

Pràctica 1

Curs 2021-22

EEstudiants: Matias Ariel Larrosa Babío

Pau Reverte Garcia

P Pablo Arana

Juve

PProfessor/a: Ramon Castells Amat

DData de lliurament: 18/03/2022

Joc de proves

Prova	Numero Aleatori	Força bruta	Estratègia optima	Estratègia taula	N_Vegades
1	Numero aleatori	Temps mitja força bruta: 0.000350	Temps mitja estratègia optima: 0,000000	Temps mitja estratègia taula: 0.000550	10
2	Numero aleatori	Temps mitja força bruta: 0.000250	Temps mitja estratègia optima: 0,000000	Temps mitja estratègia taula: 0.000700	10
3	Numero aleatori	Temps mitja força bruta: 0.000350	Temps mitja estratègia optima: 0,000000	Temps mitja estratègia taula: 0.000350	10
4	Numero aleatori	Temps mitja força bruta: 0.000285	Temps mitja estratègia optima: 0.000004	Temps mitja estratègia taula: 0.000452	1000
5	Numero aleatori	Temps mitja força bruta: 0.000307	Temps mitja estratègia optima: 0.000004	Temps mitja estratègia taula: 0.000513	1000
6	Numero aleatori	Temps mitja força bruta: 0.000278	Temps mitja estratègia optima: 0.000003	Temps mitja estratègia taula: 0.000442	1000
7	Numero aleatori	Temps mitja força bruta: 0.000287	Temps mitja estratègia optima: 0.000003	Temps mitja estratègia taula: 0.000700	10000
8	Numero aleatori	Temps mitja força bruta: 0.000286	Temps mitja estratègia optima: 0.000003	Temps mitja estratègia taula: 0.000453	10000
9	Numero aleatori	Temps mitja força bruta: 0.000288	Temps mitja estratègia optima: 0.000003	Temps mitja estratègia taula: 0.000465	10000

Cost de memòria

Per al cost de memòria utilitzem una recursivitat substractiva i tenint en compte el nostre codi el cost inicial de la memòria serà on Cb(n) on és O(1)i Cnr(n2), el número de crides és 3, per tant, a=3 i b serà igual a 1ja que serà el subproblema que ens donarà major cost si ens posem en el pitjor cas, per tant, tenint en compte la fórmula extreta de teoria:

Recursivitat subtractiva

$$T(n) = \begin{cases} O(n^{k+1}) & a = 1 \\ O(a^{n \text{ div } b}) & a > 1 \end{cases}$$

Imatge extreta del pdf de troria

En aquest cas, T(n)=3n2/1, doncs, és igual a T(n)=3n2
L'ordre essent així O(n^2).
Si ens posem a fer els costs de les tres estratègies per separat seria:

-Força bruta: com que hi ha el bucle for ens donarà que O(n) essent aquest l'orde major dins de la funció.

-Òptima: no sabem ven bé que cost té, sabem que dependrá de el cost de la funció sqrt() si es constant el cost de la funció es O(1) si no ho es, serà el cost de la funció de sqrt O(O(sqrt()). En qualsevol cas ens inclinem a pensar que té un cost més aviat constant ja que a les proves té en la gran majoria un temps de 0s.

-Taula: l'estratègia de la taula podem veure que recorrem la taula fins a trobar l'element o fins a comprovar que l'element de la taula actual es més gran que x (com estan en ordre més endavant no estarà). Com que les posicions dels números triangulars segueixen la forma de $\frac{(\sqrt{8x+1}-1)}{2}$recorrerem la taula aquest nombre de vegades. Per tant, O($\sqrt{n}$)

Per tant, podrem veure que tal com passa a la taula de joc de proves en la gran majoria dels casos l'estratègia òptima és la millor quant a temps i en conseqüència cost i la de la taula és la pitjor per poca diferència amb la de la força bruta, però al tenir que pre computar els 50 000 primers nombres triangulars i guardar-los seria la pitjor estratègia.