

**Sistemes operatius – Curs 19-20**

**Grau d’Enginyeria Informàtica**

**Departament de Matemàtiques i Informàtica**

**Facultat de Matemàtiques i Informàtica**

PROBLEMES SESSIÓ 07

Eficiencia Read i fRead

**Jorge Vinagre Triguero**

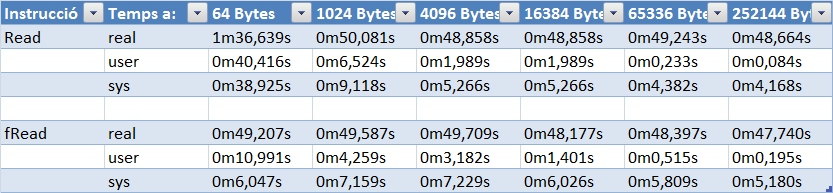
**Lluís Roca Roman**

**Grup B00**

**24/04/2020**

Resultats:

Els següents resultats han sigut obtinguts de l’execució amb sistema ubuntu (linux natiu, disc magnètic, caché buidada per comanda linux):



Després de fer les comprovacions i d’obtenir els resultats, ens hem adonat que al utilitzar la trucada al io system\_call (read/write()) es triga més temps en llegir tot l’arxiu que no pas amb la comanda fread/fwrite() que és més eficient.

Les diferències que hem detectat entre la funció read()/write() i fread()/fwrite() són que el read()/write() no utilitza memòria intermitja i es fa una crida directa al sistema, en canvi el fread()/fwrite() forma part de la biblioteca de c i s’ha de trucar a la funció read()/write() per omplir el seu buffer intern. El fread()/fwrite() s’utilitza normalment per accedir a dispositius de bloc i el read()/write() per a dispositius de caràcters.

Hem observat que a mesura que augmentem la mida del bloc de bytes el temps real disminueix i el temps d’usuari i de sistema també. En quant al buffer més petit, veiem que el read triga bastant més i creiem que és degut a que al necesitar accedir més vegades a aquest buffer i al no tindre un buffer intern(com en fRead/fwrite()) sinó compartit, el procés és més lent.

També veiem que com més gran és el buffer el temps d’execució en mode usuari

i en mode sistema és menor, però el temps d’execució real, malgrat que també disminueix, no varia gaire. Pensem que això succeeix perquè el temps real inclou tots els moments en el que el procés està *sleeping,* està esperant per accedir a disc, etc…, o esperant a que acabin altres processos. En canvi, el user y el sys no inclouen aquests temps en els que el procés està inactiu. Igualment veurem que han trigat temps però serà degut al moviment de dades entre els buffers.