**SISTEMES OPERATIUS II: Pràctica 3**

**1) Expliqueu què vol dir que una arquitectura pugui ser big-endian o little-endian.**

Partirem explicant que significa el format endian. Aquest es el que designa el format en que s’emmagatzema les dades de més d’un byte a l’ordinador. L’arquitectura vindrà donada segons l’ordinador on s’ha escrit les dades, per exemple l’arquitectura big-endian es adoptat per Motorola, mentre que l’arquitectura little-endian es adoptada per Intel.

Que una arquitectura sigui big-endian ve referida a que els bytes es representen del més significatiu al menys significatiu.

En canvi que una arquitectura sigui little-endian seria tot el contrari, els bytes es representen del menys significatiu al més significatiu.

Un exemple representatiu seria el següent:

* Tenint el numero hexadecimal : 0x3133
* BIG-ENDIAN : 0x31, 0x33
* LITTLE-ENDIAN : 0x33 , 0x31

**2) Comenteu també quines implicacions té a l’hora de emmagatzemar o carregar dades de disc.**

La implicació no vindria si s’emmagatzema o es carrega en un mateix ordinador, si no si es realitzen entre diferents, ja que si un ordinador tingues arquitectura big-endian i l’altre ordinador tingues arquitectura little-endian, el fitxer es llegiria al reves i no tindria cap sentit.

**3)Quines implicacions té per a cadascuna de les dades que vosaltres emmagatzemeu a disc?**

La implicació ve donada en que nosaltres hem guardat les dades seguint un ordre concret.

Primer guardem el màgic number, després el numero de nodes de l’arbre i finalment guardem el conjunto del node, amb la longitud de la paraula, la paraula i el numero de vegades que hi apareix.

En el moment en que es canvies l’arquitectura, per exemple llegir les dades des d’un ordinador que tingues una arquitectura diferent al que l’hem realitzat, els bytes es llegirien ordre invers i no seguiria el format demanat a l’enunciat, per la qual cosa no tindria sentit la sortida del programa.

**4) Què cal fer per assegurar que l’arbre que emmagatzemeu a disc sigui compatible entre arquitectures big-endian i little-endian?**

Per assegurar que l’arbre sigui compatible entre les dues arquitectures s’haurien d’intercanviar l’ordre dels bytes. Aquesta manera d’anar intercanviant l’ordre dels bytes pot arribar a no ser perfecta del tot ja que pot ser que es canviïn alguns bytes que no son necessaris canviar. Una manera més simple de fer-ho seria convertir totes les dades en cadenes de caràcters ASCII que son independent de l’arquitectura endiana. Per exemple en comptes de enviar 0x010A, el programa enviaria la cadena ASCII de 3 bytes “266” ja que com no depenen del l’ordre, l’altra maquina l’entendria i convertiria l’ASCII al format enter natiu de la pròpia maquina.