Ejercicio 1:

1) a) Para un tamaño de memoria física de 4GB y direccionamiento (espacio que ocupa una unidad de direccionamiento) de 1B vemos que # unidades de direccionamiento (celdas de memoria) es $\frac{4GB}{1B} \cdot \frac{2^{30}B}{1GB} = 4 \cdot 2^{30} = \underline{2^{32}}$. Luego para distinguirlas se necesita una dirección de memoria de al menos $\log_2(2^{32}) = \underline{32 \text{ bits}}$

b) Tamaño de la memoria física = 8GB, direccionamiento = "media palabra" = 16 bits (una palabra son 32 bits) = 2B (8 bits = 1B) $\Rightarrow$ # celdas = $\frac{8GB}{2B} \cdot \frac{2^{30}B}{1GB} = 4 \cdot 2^{30} = \underline{2^{32}}$. Con esto, # bits = $\log_2(2^{32}) = \underline{32}$ (la dirección de memoria debe tener al menos 32 bits)

c) Tamaño de la memoria física = 16GB; direccionamiento = 1 palabra = 32 bits $\cdot \frac{1B}{8 \text{ bits}} = 4B \Rightarrow$ # celdas = $\frac{16GB}{4B} \cdot \frac{2^{30}B}{1GB} = 4 \cdot 2^{30} = \underline{2^{32}}$. Luego # bits = $\log_2(2^{32}) = \underline{32}$

d) Tamaño de la memoria física = 32GB; direccionamiento = "palabra doble" = 64 bits $\cdot \frac{1B}{8 \text{ bits}} = 8B \Rightarrow$ # celdas = $\frac{32GB}{8B} \cdot \frac{2^{30}B}{1GB} = 4 \cdot 2^{30} = \underline{2^{32}}$. Así, # bits = $\log_2(2^{32}) = \underline{32}$

Ejercicio 2:

2) a) Siendo que la arquitectura posee palabras e instrucciones de y B (bytes), una memoria física de x B y direccionamiento a palabra (y B), luego veo que # celdas = $\frac{xB}{yB} = \frac{x}{y}$ y luego la cantidad de bits necesarios para distinguirlas en una dirección es techo$(\log_2(\frac{x}{y}))$ = $\underline{\text{techo}(\log_2(x) - \log_2(y))}$ (donde techo$(x) = \lceil x \rceil = \min\{k \in \mathbb{Z}, k \geq x\}$)

Nótese que no tomamos en cuenta el caso en que # celdas < 1 ya que en dicha circunstancia la memoria no tendría celdas en las cuales almacenar datos.

b) Siendo que las instrucciones son de y B = $8 \cdot y$ bits y cada una posee un operando con modo de direccionamiento a memoria, esto nos dice que de los $8 \cdot y$ bits se usarán $\lceil \log_2(x) - \log_2(y) \rceil$ bits para una dirección de memoria. Esto nos deja con $\underline{8y - \lceil \log_2(x) - \log(y) \rceil}$ bits disponibles para el código de operación. Como los bits sólo pueden valer 0 o 1, el número máximo de códigos de operación es $\underline{2^{8y - \lceil \log_2(x) - \log_2(y) \rceil}}$

c) Si ahora $x = 2^k$ y además $y = 2^j$ donde $k, j \in \mathbb{N}$ (x e y son potencias de dos), vemos que ahora la cantidad de bits necesarios para distinguir todas las celdas de la memoria es $\lceil \log_2(2^k) - \log_2(2^j) \rceil = \lceil k - j \rceil = \underline{k - j}$ (pues $k, j \in \mathbb{N}$). A su vez, la máxima cantidad posible de códigos de operación es $2^{8 \cdot 2^j - (k-j)} = \underline{2^{2^{j+3} - (k-j)}}$

d) La máxima granularidad de acceso a memoria se obtiene cuando la unidad direccionable es $\underline{\text{lo más chica posible}}$. Al ser el direccionamiento de la arquitectura a palabra y una palabra tiene 2^j bytes, para ser lo más chica posible j debe ser $\underline{0}$, así la unidad direccionable es de $2^0 = \underline{1 \text{ byte}}$, lo cual hace que para un tamaño de memoria dado la cantidad de celdas sea máximo.