Dadas la siguientes funciones en C++:

```
char *strupr (char *s) {
   char *mayusculas="ABCDEFGHIJKLMNOPQRSTUVWXYZ" ;
   char *minusculas   ="abcdefghijklmnopqrstuvwxyz" ;

   for (int i=0; i<strlen(s); i++) {
      for (int j=0;j<strlen(minusculas);j++)
         if (s[i]==minusculas[j])
            s[i]=mayusculas[j] ;
   return s ;
}

int strlen (char *s) {
   int i=0 ;
   while ( s[i] != 0 )
      i++ ;
   return i ;
}
```

a. Deduzca formalmente O(n) , mostrando claramente su procedimiento
b. Convierta el algoritmo anterior a un algoritmo equivalente (mismos resultados) pero con O(n)=n

202100081-Javier M onjes

$$T(n) = \begin{cases} T(cond) + T(return) = 2t & si\ n \leq 1 \\ T(cond) + T(return) + T\left(\frac{n}{2}\right) + T\left(\frac{n}{2}\right) = 2t + 2T\left(\frac{n}{2}\right) & si\ n > 1 \end{cases}$$

$$T(n) = 2t + 2T\left(\frac{n}{2}\right)$$

$$= 2t + 2\left(2t + 2T\left(\frac{n}{4}\right)\right) = (2+4)t + 4T\left(\frac{n}{4}\right)$$

$$= (2+4)t + 4\left(2t + 2T\left(\frac{n}{8}\right)\right) = (2+4+8)t + 8T\left(\frac{n}{8}\right)$$

$$= (2+4+8)t + 8\left(2t + 2T\left(\frac{n}{16}\right)\right) = (2+4+8+16)t + 16T\left(\frac{n}{16}\right)$$

$$T(n) = \left(2 + 2^2 + 2^3 + \cdots 2^k\right)t + 2kT\left(\frac{n}{2^k}\right)$$

$$Si\ T(1) = 2t$$

$$\frac{n}{2^k} = 1 \to n = 2^k \to \log_2 n = k$$

$$T(n) = \left(2 + 2^2 + 2^3 + \cdots 2^{\log_2 n}\right)t + 2 * (\log_2 n)T\left(\frac{n}{2^{\log_2 n}}\right) \to \left(2 + 2^2 + 2^3 + \cdots 2^{\frac{n}{2}}\right)t + 4(\log_2 n)t$$

$$O\big(T(n)\big) = O((2 + 2^2 + 2^3 + \cdots n) + 4(\log_2 n)\ t)$$

$$O\big(T(n)\big) = \max\big((2 + 2^2 + 2^3 + \cdots n), 4(\log_2 n)\big) = O(n) = n$$

b.

```
int recursiva (int n) {
   if (n <= 1)
      return 23;
   else
      return 2 * recursiva (n / 2);
}
```

$$T(n) = \begin{cases} T(cond) + T(return) = 2t & si\ n \leq 1 \\ T(cond) + T(return) + 2T\left(\frac{n}{2}\right) = 2t + T2\left(\frac{n}{2}\right) & si\ n > 1 \end{cases}$$

$$T(n) = 2t + 2T\left(\frac{n}{2}\right)$$

$$= 2t + 2\left(2t + 2T\left(\frac{n}{4}\right)\right) = (2+4)t + 4T\left(\frac{n}{4}\right)$$

$$= (2+4)t + 4\left(2t + 2T\left(\frac{n}{8}\right)\right) = (2+4+8)t + 8T\left(\frac{n}{8}\right)$$

$$= (2+4+8)t + 8\left(2t + 2T\left(\frac{n}{16}\right)\right) = (2+4+8+16)t + 16T\left(\frac{n}{16}\right)$$

$$T(n) = \left(2 + 2^2 + 2^3 + \cdots 2^k\right)t + 2kT\left(\frac{n}{2^k}\right)$$

$$Si\; T(1) = 2t$$

$$\frac{n}{2^k} = 1 \rightarrow n = 2^k \rightarrow \log_2 n = k$$

$$T(n) = \left(2 + 2^2 + 2^3 + \cdots 2^{\log_2 n}\right)t + 2 * (\log_2 n)T\left(\frac{n}{2^{\log_2 n}}\right) \rightarrow \left(2 + 2^2 + 2^3 + \cdots 2^{\frac{n}{2}}\right)t + 4(\log_2 n)t$$

$$O\big(T(n)\big) = O((2 + 2^2 + 2^3 + \cdots n) + 4(\log_2 n)\; t)$$

$$O\big(T(n)\big) = \max\big((2 + 2^2 + 2^3 + \cdots n), 4(\log_2 n)\big) = O(n) = \log_2 n$$