

THE BEATLES

Tarefa 02 - Leonardo Rethier scores Cardoso

1. a)

```

void unificarLista (int tamLista, Lista lista) {
    int a, b;
    For (int i = 0; i < (tamLista - 1); i++) {
        a = lista.remove(1);
        b = lista.remove(2);
        int valor = a + b;
        bool menor = true;
        bool inserido = false;
        Celula* count = lista.primario; // Celula
        int count2 = 0; // posicao
        while (menor && (count != NULL)) {
            if (count->elemento > valor) {
                menor = false;
                lista.inserir(count2, valor);
                inserido = true;
            }
            count2++;
        }
        if (!inserido) {
            lista.inserirFinal(valor);
        }
    }
}

```


DATA

STQQSSD

THE BEATLES

d) alha pegando o melhor caso que é $O(n)$, com $T(n) = n - 1$ é possível, mas em seu pior caso temo $O(n^2)$ tornando impossível um tempo sublinear.



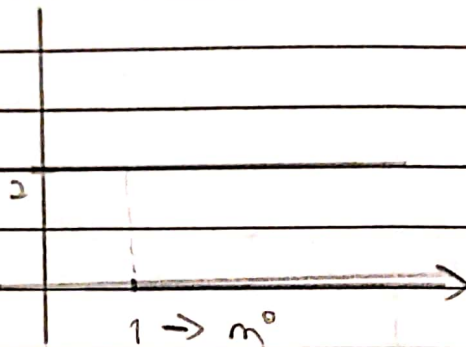
$$2-a) 2^{n+1} = O(2^n)$$

$$2^{n+1} \leq (2^n) \cdot 2, \forall n \geq n_0$$

$$2 \leq (2^n) \cdot 2, \forall n \geq n_0$$

$$1 \leq (2^n) \cdot 2, \forall n \geq n_0$$

$$f(n) = 2$$



$$2 \leq 2, \forall n \geq 1 \quad \text{e' verdade}$$

$$2^{n+1} = O(2^n)$$

THE
BEATLES

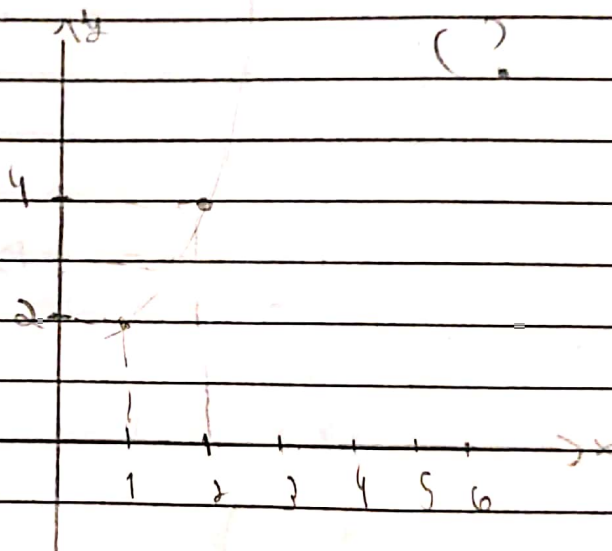
$$b) 2^{2^n} = O(2^n)$$

$$2^{2^n} \leq C 2^n, \quad \forall n \geq n_0$$

$$2^{2^n} < C 2^n, \quad \forall n \geq n_0$$

$$2^{2^n} \leq C, \quad \forall n \geq n_0$$

$$f(n) = 2^n$$



Como não existe, logo Θ impossível serem iguais.
 Prova.



$$c) f(n) = \Theta(u(n)) \text{ e } g(n) = \Theta(v(n)) \Rightarrow f(n) + g(n) = \Theta(u(n) + v(n))$$

→ pela quinta propriedade:

$$f(n) + g(n) = \Theta(\max(u(n), v(n)))$$

Verdadeiro. (Considerando que $u(n) + v(n) = w(n)$, $w(n)$ tem uma constante que multiplica a variável de maior custo, não afetando sua complexidade, por exemplo: $u(n) = n^2 + n^2 + \dots$, $v(n) = n^4 + 3n^2$; neste caso, o valor relevante da soma seria n^4 , tendo assim $\Theta(n^4)$, da mesma forma que $\Theta(\max(u(n), v(n)))$ seria $\Theta(n^4)$).