Prática 1

Prob 4-1

- $T(n) = 2T(n/2) + n^4$

  $a = 2$

  $b = 2$ $\quad \log_b a = 1 < d \quad T(n) = O(n^4)$

  $d = 4$

- $T(n) = T(7n/10) + n$

  $a = 1$

  $b = 10/7$ $\quad \log_b a = 0 < d \quad T(n) = O(n)$

  $d = 1$

- $T(n) = 16T(n/4) + n^2$

  $a = 16$

  $b = 4$ $\quad \log_b a = \log_4 16 = 2 = d \quad T(n) = O(n^2 \log n)$

  $d = 2$

- $T(n) = 7T(n/3) + n^2$

  $a = 7$

  $b = 3$ $\quad \log_b a = \log_3 7 < 2$

  $d = 2$

  $T(n) = O(n^2)$

- $T(n) = 7T(n/2) + n^2$

  $a = 7$

  $b = 2$ $\quad \log_b a = \log_2 7 > 2$

  $d = 2$

  $T(n) = O(n^{\log_2 7})$

- $T(n) = 2T(n/4) + \sqrt{n}$

  $a = 2$

  $b = 4$ $\quad \log_b a = \log_4 2 = 1/2 = d$

  $d = 1/2$

  $T(n) = O(\sqrt{n} \cdot \log n)$

- $T(n) = T(n-2) + n^2$

$$T(n) = n^2 + T(n-2)$$
$$= n^2 + (n-2)^2 + T(n-4)$$
$$= n^2 + (n-2)^2 + (n-4)^2 + T(n-6)$$
$$= n^2 + (n-2)^2 + (n-4)^2 + (n-6)^2 + T(n-8)$$
$$= \sum_{i=0}^{?} (n-2i)^2 \longrightarrow \sum_{i=0}^{n/2} (n-2i)^2$$

$2k = n \Leftrightarrow k = n/2$

$$= \sum_{i=0}^{n/2} (n^2 + 4i^2 - 4in)$$
$$= n^2 \sum_{i=0}^{n/2} 1 + 4 \sum_{i=0}^{n/2} i^2 - 4n \sum_{i=0}^{n/2} i$$
$$= O(n^3)$$

Prob 4-3.

- $T(n) = T(n-1) + \log n$

$$
\begin{aligned}
T(n) &= \log n + T(n-1) \\
&= \log n + \log n-1 + T(n-2) \\
&= \log n + \log n-1 + \log n-2 + \cdots + \overset{=0}{\log 1} \\
&\vdots \\
&= \log n \cdot (n-1) \cdots 1 \\
&= \log n!
\end{aligned}
$$

$$H_n = \underbrace{1 + 1/2 + 1/3 + \cdots + 1/n}_{\text{upper bound?}}$$

- $T(n) = T(n-1) + 1/n$

$$T(n) = 1/n + T(n-1)$$
$$= 1/n + 1/n-1 + T(n-2)$$
$$= \underbrace{1/n + 1/n-1 + 1/n-2 + \cdots}_{\text{série harmónica}}$$
$$\geq \log_2(n+1)$$

$$H_n = 1 + 1/2 + 1/3 + 1/4 + 1/5 + 1/6 + 1/7 + 1/8 + \cdots + 1/n$$
$$\leq 1 + \underbrace{1/2 + 1/2}_{1} + \underbrace{1/4 + 1/4 + 1/4 + 1/4}_{1} + \underbrace{1/8 + \cdots + 1/8}_{1} + \underbrace{1/16 + \cdots}_{1} + 1/n$$

$$H_{31} \leq 1 + 1 + 1 + 1 = 4$$

$$H_n \leq \underbrace{1 + \cdots + 1}_{k \text{ vezes}}$$ quem é k?

$k=3$ $$1 + \overbrace{1/2 + 1/2}^{2} + \overbrace{1/4 + \cdots + 1/4}^{4}$$

- Qts termos aparecem na soma se o resultado é k?

$$\sum_{i=0}^{k-1} 2^i = 2^k - 1$$

$$n \leq 2^k - 1$$
$$n+1 \leq 2^k$$
$$k \geq \log_2(n+1)$$

$$\sum_{i=0}^{n} i^2 = ?$$

$$\sum_{i=1}^{n} \left(i^3 - (i-1)^3 \right) = n^3$$

$$\sum_{i=1}^{n} i^3 - \left(i^3 - 3i^2 + 3i - 1 \right) = n^3$$

$$\sum_{i=1}^{n} 3i^2 - \sum_{i=1}^{n} 3i + 1 = n^3$$

$$\sum_{i=1}^{n} 3i^2 - 3\sum_{i=1}^{n} i + \sum_{i=1}^{n} 1 = n^3$$

$$3 \cdot \sum_{i=1}^{n} i^2 = n^3 + 3 \cdot \frac{n(n+1)}{2} - n$$

$$\sum_{i=1}^{n} i^2 = \frac{n^3}{3} + \frac{n(n+1)}{2} - \frac{n}{3}$$

$$= \frac{2n^3 + 3n^2 + 3n - 2n}{6}$$

$$= \frac{2n^3 + 3n^2 + n}{6}$$

$$= \frac{n(2n^2 + 3n + 1)}{6}$$

$$= \frac{n(2n+1)(n+1)}{6}$$

TI de 13/14

```
int f(int n)
{ int i = 0;
  while (i < n)
    i++
  if (n > 1)
    i = 2 * f(n/4) + f(n/4)
  return i
}
```

→ nº de iterações do loop

k	i
0	0
1	1
2	2
k	k

$i = n$

$k = n$

$$T(n) = 2\,T(n/4) + O(n)$$

$a = 2$

$b = 4$

$d = 1$

$\log_b a = 1/2 < d$

$T(n) = O(n)$

R 13/14

```
int f (int n) {
    int j, i;
    i = 0;
    j = 0;
    while (i < n) {
        j++;
        i += 2
    }
    if (n > 1) {
        i = 2 * f(j) + f(j)
    }
    return i
}
```

Quantas iterações?

k	i	j
0	0	0
1	2	1
2	4	2
⋮		
k	2k	k

Condição de paragem: $i = n$

$2k = n$

$k = n/2$ → $j = n/2$

$$T(n) = 2T(n/2) + O(n)$$

$a = 2$

$b = 2$ $\quad \log_b a = 1 = d$ $\quad T(n) = O(n \log n)$

$d = 1$

TI 14/15 I.e

```
int f(int n) {
    int i = 0;
    while (i*i < n) {
        i++
    }
    if (n>1)
        i = f(n/4) + f(n/4) + f(n/4)
    return i
}
```

Quantas iterações?

k	i
0	0
1	1
2	2
⋮	
k	k

Condição de paragem:

$i * i = n$

$k^2 = n$

$k = \sqrt{n}$

$$T(n) = 3\,T(n/4) + O(\sqrt{n})$$

$a = 3$

$b = 4$

$d = 1/2$

$\log_b a = \log_4 3 \bigcirc 1/2$

$4^{1/2} = 2 < 3$

$\log_4 3 > 1/2$

$$T(n) = O(n^{\log_4 3})$$

TI 16/17 - I.e

```
int f (int n) {
   int i=0, j=0;
   while (n*n > i) {
      i = i+2
      j++
   }
   if (n>1)
      i = 5* f(n/2) + f(n/2) + f(n/2) + f(n/2);
   while (j>0) {
      i = i+2
      j--
   }
   return i
}
```

k	i	j
0	0	0
1	2	1
2	4	2
3	6	3
k	2k	k

j final $\frac{n^2}{2}$

Condição de paragem:

$2k = n*n$

$k = \frac{n*n}{2}$ $\Rightarrow$ nº de iterações $\underline{\underline{O(n^2)}}$

$$T(n) = 4\,T(n/2) + O(n^2)$$

$a = 4$

$b = 2$

$d = 2$

$\log_b a = 2 = d$

$T(n) = O(n^2 \log n)$

R1 16/17

```
int f(int n) {
    int i=0, j=0;
    while (j<10) {
        i = i+2
        j++
    }
    if (n>1)
        i += f(n/2) + 3f(n/2)
    while (j>0) {
        i--;
        j = j-2
    }
    return i
}
```

$T(n) = 2T(n/2) + O(1)$

$\log_b a = 1 > 0$

$T(n) = \underline{O(n)}$

EI 18/19 Ie

$$T(n) = 4\,T(n/2) + O(\log n)$$

$\log_b a = \log_2 4 = 2$ $\qquad (\log n) \in O(n^{2-\varepsilon})$

$$T(n) \in O(n^2)$$