

Esercitazione 23-04-2020

Word count: 471

Esercizio 1

m1 è la radice
 m2 è figlio della radice
 m3 o è figlio di m2 o è figlio m1
 m4 figlio di m3 o m2 o di m1
 il quarto valore min è tra i primi $1+2+4+8$ elementi dell'array.

```
fun extract4min(H, n) {
    if (n<4) return null           c1

    m = min{15, n}
    A = H[1...m] // primi 15 elementi in H      c2 * 15
    sort(A, m)                                O(15 log 15)
    return A[4] // quarto elemento            c3
}
```

$$T(n) = c_1 3 + 15 * c_2 + O(1) = \Theta(1)$$

Esercizio 2

```
void function(n) {
    count = 0;                      c1
    for (i=n/2; i<=n; i++)         n/2-volte
        for (j=1; j<=n; j = 2 * j)   ( $\log_2 n$ ) -volte
            for (k=1; k<=n; k = k * 2)   ( $\log_2 n$ )-volte
                count++;                  c2
}
```

$$j'=0.. \log_2 n \quad j=2^{j'}$$

$$k'=0.. \log_2 n \quad k=2^{k'}$$

$$\begin{aligned} T(n) &= c_1 + \sum_{i=n/2..n} [\sum_{j'=0..\log_2 n} [\sum_{k'=0..\log_2 n} [c_2]]] = \\ &= c_1 + \sum_{i=n/2..n} [\sum_{j'=0..\log_2 n} [c_2 \log_2 n]] = \\ &= c_1 + \sum_{i=n/2..n} [c_2 * \log_2 n * \log_2 n] = \\ &= c_1 + c_2 * n/2 * \log_2 n * \log_2 n = \end{aligned}$$

$$= c1 + c2/2 n \log_2^2 n = \Theta(n \log^2 n)$$

$$\log^2 n = (\log n * \log n)$$

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

Esercizio 3

Soluzione 3.1

Scorro A, per ogni a_i cerco in B a_i+3 oppure a_i-3 , se lo trovo ritorno la coppia di indici, altrimenti al termine ritorno (-1, -1).

```
fun ex3(A, n, B, m) {
    for i=0..n-1 {                                n-volte
        j = binSearch(B, A[i]+3, 0, m-1)          O(log m)
        if (j != -1) return (i, j)                  c1
        j = binSearch(B, A[i]-3, 0, m-1)          O(log m)
        if (j != -1) return (i, j)                  c2
    }
    return (-1, -1)                            c3
}
```

Analisi.

$$T(n) = O(n \log m)$$

Se $m < n$ allora posso inizialmente scambiare A e B per ottenere $O(m \log n)$.

Se sono circa uguali $n = \Theta(m)$ allora $T(n) = O(n \log n)$.

Soluzione 3.2

Modifico la funzione merge per scorrere parallelamente A e B cercando $a_i+3 = b_j$. Se non lo trovo farò la stessa cosa cercando $a_i-3 = b_j$.

```
fun ex3(A, n, B, m) {
    j = 0
    for i=0..n-1
        while ((j < m) && (B[j] < A[i]+3)) j++
        if (j >= m) break
        if (B[j] == A[i]+3) return (i, j)

    for i=0..n-1
        while ((j < m) && (B[j] < A[i]-3)) j++
        if (j >= m) break
        if (B[j] == A[i]-3) return (i, j)

    return (-1, -1)
```

}

$$T(n) = O(n) + O(m) = O(n+m)$$

Se sono circa uguali $n=\Theta(m)$ allora $T(n) = O(n)$.

1234567890 Saverio Caminiti