

```
/*
Carlos Eduardo Gonzaga Romaniello de Souza - 19.1.4003
*/

/*
A)
Maior valor em um vetor
*/

/*

Análise assintótica

 $T(1) = 1$ 
 $T(n) = 2T(n/2) + 4 = 2T(n/2) + O(1)$ 

Pelo teorema mestre simplificado temos que:  $a = 2$ ,  $b = 2$ ,  $d = 0$ 
 $\log_2(2) > 0$ 

 $\text{getMax}() = O(n)$ 

*/
void getMax(int* v, int esq, int dir, int* max) {
    if (dir - esq <= 0) {return;} // 1

    int meio = floor((dir + esq) / 2); // 1
    if (v[meio] > *max) {*max = v[meio];} // 2

    getMax(v, esq, meio, max); // T(n/2)
    getMax(v, meio + 1, dir, max); // T(n/2)
}

/*
B)
Maior e menor valor em um vetor
*/

/*

Análise assintótica

 $T(1) = 1$ 
 $T(n) = 2T(n/2) + 6 = 2T(n/2) + O(1)$ 

Pelo teorema mestre simplificado temos que:  $a = 2$ ,  $b = 2$ ,  $d = 0$ 
 $\log_2(2) > 0$ 

 $\text{getMaxMin}() = O(n)$ 
```

```

*/
void getMaxMin(int* v, int esq, int dir, int* max, int* min) {
    if (dir - esq <= 0) {return;} // 1

    int meio = floor((dir + esq) / 2); // 1
    if (v[meio] > *max) {*max = v[meio];} // 2

    if (v[meio] < *min) {*min = v[meio];} // 2

    getMaxMin(v, esq, meio, max, min); // T(n/2)
    getMaxMin(v, meio + 1, dir, max, min); // T(n/2)
}

```

```

/*
C)
Exponenciacao
*/

```

```

/*

```

Análise assintótica (sem considerar operações aritmeticas sendo bit a bit)

$T(1) = 1$   
 $T(n) = T(n/2) + 6 = T(n/2) + O(1)$

Pelo teorema mestre simplificado temos que:  $a = 1$ ,  $b = 2$ ,  $d = 0$   
 $\log_2(1) = 0$

$\text{exponenciacao}() = O(\log(n))$

```

*/
int exponenciacao(int base, int pot) {
    if (pot == 0) return 1; // 1
    int resultado = exponenciacao(base, pot / 2); // t(n/2)

    if (pot % 2 == 0) {return resultado * resultado;} // 2
    else {return base * resultado * resultado;} // 3
}

```