```
void maxMin1(int *array, int tamArray, int *max, int *min){

    *max = array[0];                    1
    *min = array[0];                    1

    for(int i=1; i<tamArray; i++){      1 + n + (n-1)
        if(array[i] > *max){            n-1
            *max = array[i];            n-1
        }
        if (array[i] < *min){           n-1
            *min = array[i];            0
        }
    }
}
```

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

T(n) = 5n-1

==========

```
void maxMin2(int *array, int tamArray, int *max, int *min){

    *max = array[0];                    1
    *min = array[0];                    1

    for(int i=1; i<tamArray; i++){      1 + n + (n-1)
        if(array[i] > *max){            n-1
            *max = array[i];            0
        } else if (array[i] < *min){    n-1
            *min = array[i];            n-1
        }
    }
```

```
}
```

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

T(n) = 5n-1

==========

```
void maxMin3(int *array, int tamArray, int *max, int *min){

    if(tamArray%2 != 0){                        1
        array[tamArray] = array[tamArray-1];    1
        tamArray++;                             1
    }

    *max = array[0];                            1
    *min = array[1];                            1
    if(array[0] < array[1]){                    1
        *max = array[1];                        1
        *min = array[0];                        1
    }

    for(int i=2; i<tamArray-1; i+=2){           1 + ((n/2) - 1) + 1 + (n/2 - 1)
        if(array[i] > array[i+1]){              n/2 – 1
            if(array[i] > *max){                n/2 - 1
                *max = array[i];                n/2 - 1
            }
            if(array[i+1] < *min){              n/2 - 1
                *min = array[i+1];              n/2 -1
            }
        } else {
            if(array[i] < *min){
                *min = array[i];
```

```
            }
            if(array[i+1] > *max){
                *max = array[i+1];
            }
        }
    }
}
```

T(n) = 8 + 1+((n/2)-1)+1+(n/2 - 1)+n/2 + n/2-1 + n/2-1 + n/2-1 + n/2-1 + n/2-1

T(n) = 8n/2-3

T(n) = 4n-3