

Act 1.1.2 Análisis de algoritmos iterativos con notación asintótica

1)

a) Suma la sucesión de números elevados por sí mismos, incrementando uno por uno, desde uno, siempre y cuando i sea menor o igual a n

b) $s = s + i * i$

c) n veces

d) $O(n)$

2)

a) $O(n^3)$

b) $O(1)$

c) $O(\log n)$

d) $O(n)$

e) $O(n^2)$

f) $O(n)$

g) $O(n^2)$

h) $O(n * \log n)$

3)

a)

```
template <class T>
int iterativa(vector <T> list){
    int n = 0;
    for (int i=0 ; i<=list.size()-1 ; i++){
        if(list[i] % 2 != 0){
            n = n + list[i];
        }
    }
    return n;
}
```

$O(n)$

b)

```
template <class T>
int recursiva(vector <T> list){
    int n = 0;
    int i = 0;

    if (i>list.size()-1) {
        return n;
    } else {
        if(list[i] % 2 != 0){
            n = n + list[i];
            i++;
        }
    }
}
```

Ernesto Poisot Avila
A01734765

```
        return recursiva(list);  
    } else {  
        i++;  
        return recursiva(list);  
    }  
}  
return 0;  
}
```

$O(n)$