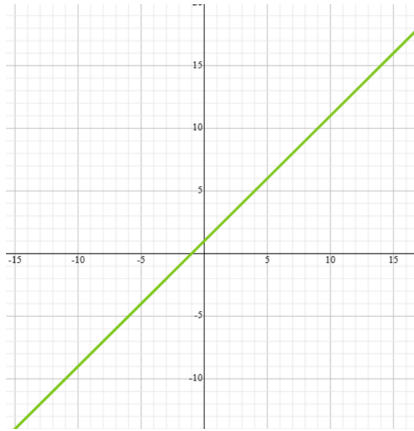
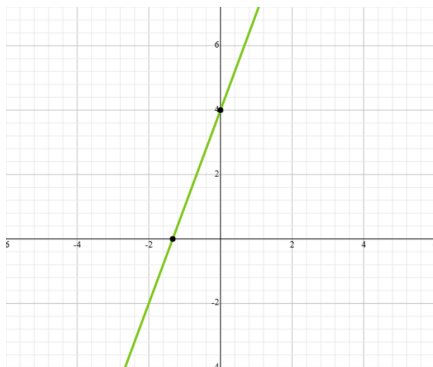


Respostas do exercício de PAA

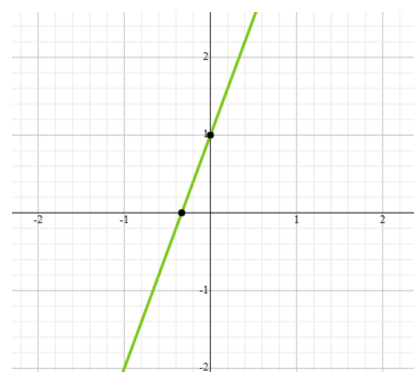
a) $n + 1$ | Gráfico: Linear | Pior caso: $O(n)$ | Melhor caso: $O(1)$



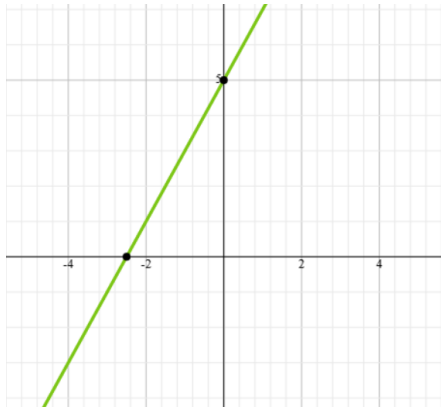
b) $3n + 4$ | Gráfico: Linear, Pior caso: $O(n)$ / Melhor caso: $O(1)$



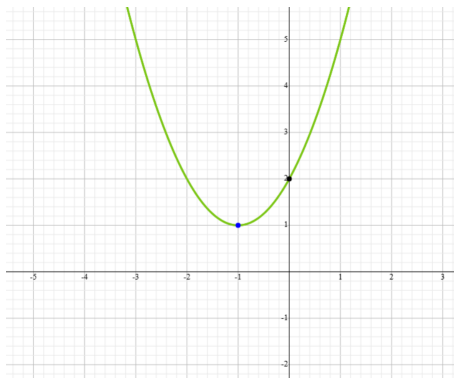
c) $3n + 1$ | Gráfico: Linear | Melhor caso: $O(n)$ | Pior caso $O(n)$



d) $2n + 5$ | Gráfico: Linear | Pior caso: $O(n)$ | Melhor caso: $O(1)$

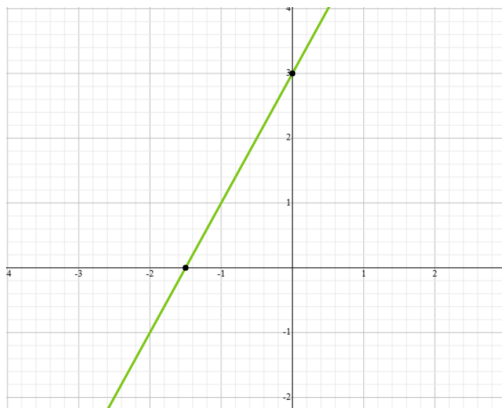


e) $n^2 + 2n + 2$ | Gráfico: Quadrático | Pior caso: $O(n^2)$ | Melhor caso $O(1)$

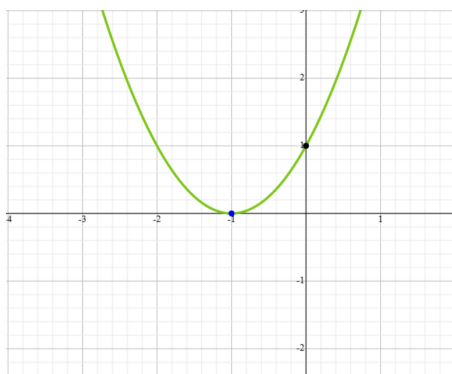


f) $4mn + 2$ | Gráfico: essa expressão não esboça um gráfico | Pior caso: $O(mn)$ | Melhor caso: $O(1)$

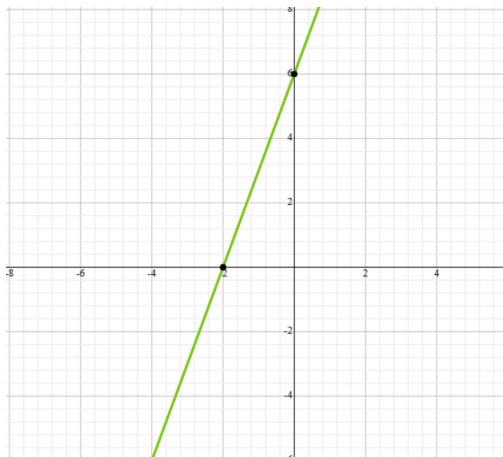
g) $2n + 3$ | Gráfico: Linear | Pior caso: $O(n)$ | Melhor caso: $O(1)$



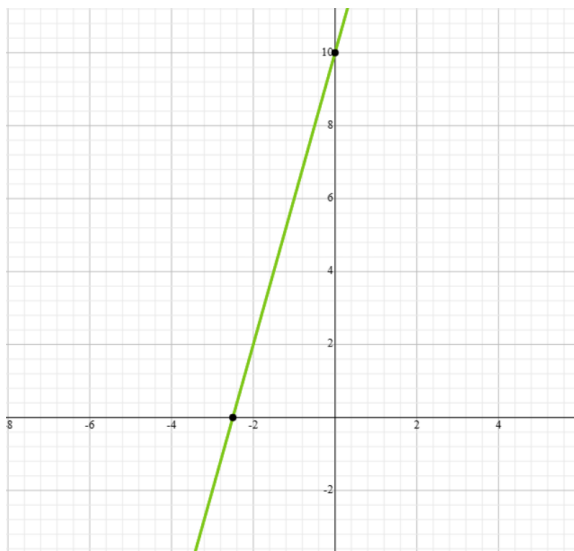
h) $n^2 + 2n + 1$ | Gráfico: Quadrático | Pior caso: $O(n^2)$ | Melhor caso: $O(1)$



i) $3n + 3$ | Gráfico: Linear | Pior caso: $O(n)$ | Melhor caso: $O(1)$



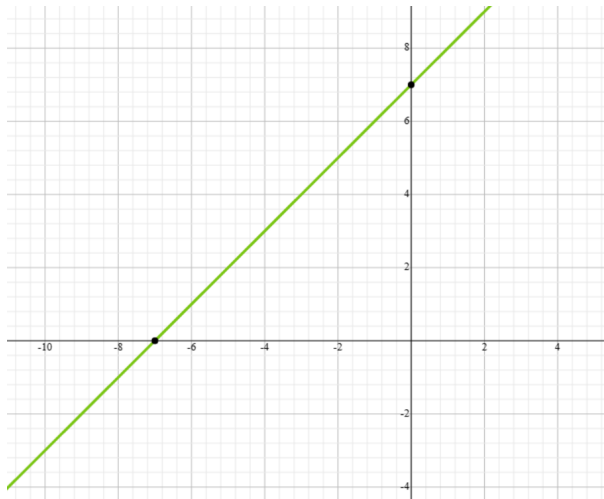
j) $4n + 10$ | Gráfico: Linear | Pior caso: $O(n)$ | Melhor caso: $O(1)$



2) Dado o método de busca a seguir, analise o tempo estimado de execução no melhor e no pior caso para cada um dos trechos de código, considerando o modelo RAM. Lembre que `size()` é um método que retorna a quantidade de elementos de uma lista.

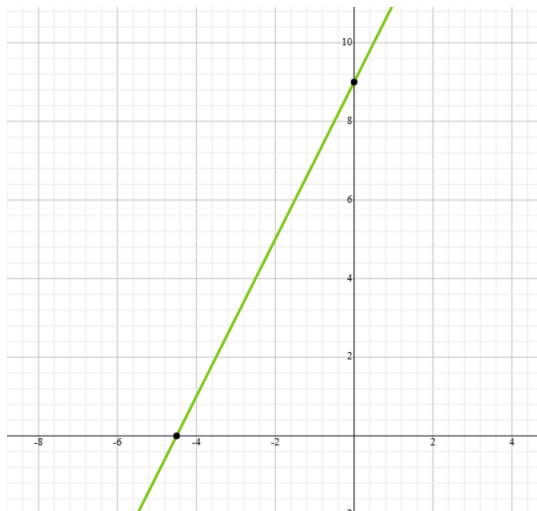
```
Pessoa busca(String nome){
    for (int i = 0; i < pessoas.size(); i++){ n + 1
        if (pessoas.get(i).getNome().equals(nome)) +1
            return pessoas.get(i); +1
    }
    return null; +1
}
```

a) $n + 7$ | Gráfico: Linear | Pior caso: $O(n)$ | Melhor Caso: $O(1)$

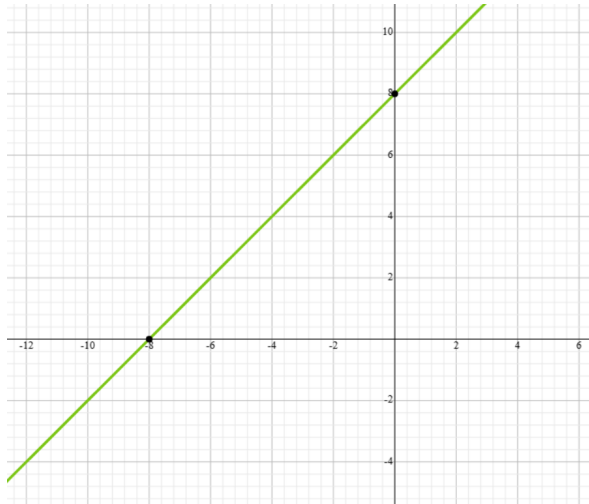


```
void exibir(String nome){
    Pessoa p = busca(nome); +1 | n + 4
    if (p != null){ +1
        p.exibirDados(); +1
    }
    else{
        System.out.println("Pessoa não encontrada");
    }
}
```

b) $2n + 9$ | Gráfico: Linear | Pior caso: $O(n)$ | Melhor caso: $O(1)$



c) $n + 8$ | Gráfico: Linear | Pior caso: $O(n)$ | Melhor caso: $O(1)$



d) $3n + 13$ | Gráfico: Linear | Pior caso: (n) | Melhor caso: $O(1)$

