

```
9) a) a = ptr head L1
      b = ptr head L2
      if (a == b) return *a
      else {
        while (a != nullptr) {
          bb = b
          while (bb != nullptr) {
            if (a == bb) return *a
            else bb = bb->next()
          }
          a = a->next()
        }
        return nullptr
      }
```

$O(n)$

$O(n^2)$

```

b) a = ptr head L1
    b = ptr head L2
    s = ptr solución
    st1, st2 = stack vacío
    if (a == b) return *a
    else {

```

```

        while (a != nullptr) {
            st1.push(a)
            a = a.getnext()
        }

```

} O(n)

```

        while (b != nullptr) {
            st2.push(b)
            b = b.getnext()
        }

```

} O(n)

```

        while (st1.top() == st2.top()) {
            s = st1.top()
            st1.pop()
            st2.pop()
        }

```

} O(n)

```

        return *s
    }

```

tiempo lineal.

c) Se veuve más fácil ya que simplemente se comentaría por el final de las listas y se iría avanzando en cada ~~elemento~~^{lista} uno por uno comparando con la otra lista, hasta que se encuentren dos nodos que sean distintos y estén a la misma distancia de la cola de la lista.