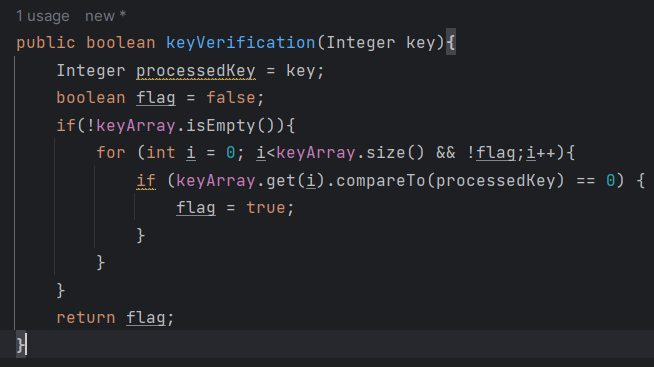


|  |  |
| --- | --- |
| Code | Time |
| Para i de 0 hasta tamaño de reminders - 1 Hacer | n+1 |
| Para j de 1 hasta tamaño de reminders - i - 1 Hacer | n2 |
| anterior <- reminders[j - 1] | n2-1 |
| actual <- reminders[j] | n2-1 |
| Si (reminders[j - 1] no es nulo) y (reminders[j] no es nulo) Entonces | n2-1 |
| Si anterior.getPriority() < actual.getPriority() Entonces | n2-1 |
| reminders[j] <- anterior | n2-1 |
| reminders[j - 1] <- actual | n2-1 |
| Total |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Tipo | Variable | Tamaño de 1 valor atómico | Cantidad de valores atomicos |
| Auxiliar | i | 32 bits | n |
| Auxiliar | j | 32 bits | n2 |
| Asignación | anterior | 112 bits | n2-1 |
| Asignación | actual | 112 bits | n2-1 |
| Total | | |  |



|  |  |
| --- | --- |
| Code | Time |
| Entero procesadaClave <- key | 1 |
| Booleano flag <- Falso | 1 |
| Si NO estáVacia(keyArray) Entonces | 1 |
| Para i desde 0 hasta tamañoDe(keyArray) - 1 Hacer | n+1 |
| Si keyArray[i] es igual a procesadaClave Entonces | n |
| flag <- Verdadero | n |
| Total | n |

|  |  |  |  |
| --- | --- | --- | --- |
| Tipo | Variable | Tamaño de 1 valor atómico | Cantidad de valores atomicos |
| Asignación | processedKey | 32 bits | 1 |
| Auxiliar | flag | 1 bit | 1 |
| Auxiliar | i | 32 bits | n |
| Total | | |  |