José Alexander Brenes Brenes.

Juan Daniel Quirós Moya.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | Arreglo int |  |  |  | Arreglo short int |  |  |
| Cantidad  de dígitos  almacenables | Tamaño del arreglo en Bytes. | Desperdicio con un solo dato. | % de bytes de memoria total de un nodo para punteros *next & prev* | Cantidad  de dígitos  almacenables | Tamaño del arreglo en Bytes. | Desperdicio con un solo dato. | % de bytes de memoria total de un nodo para punteros *next & prev* |
| 27 | 12 | 8 | 57.14 | 40 | 20 | 18 | 44.44 |
| 36 | 16 | 12 | 50.00 | 44 | 22 | 20 | 42.10 |
| 45 | 20 | 16 | 44.44 | 48 | 24 | 22 | 40.00 |
| 54 | 24 | 20 | 40.00 | 52 | 26 | 24 | 38.10 |
| 63 | 28 | 24 | 36.36 | 56 | 28 | 26 | 36.36 |
| 72 | 32 | 28 | 33.33 | 60 | 30 | 28 | 34.78 |
|  |  |  |  | 64 | 32 | 30 | 33.33 |

En nuestro proyecto decidimos utilizar arreglos de tipo int para almacenar los números de la clase Integer pues permite almacenar más del doble de elementos que el tipo de datos short con una menor cantidad de bytes, ventaja deseable cuando se trabajan números muy grandes. Además, como se observa en la tabla anterior que definimos a través pruebas, observamos una clara ventaja en el uso de la memoria pues si el valor a almacenar es mínimo (desde -99999999 hasta 999999999), el desperdicio de memoria es considerablemente menor. No se utilizó el tipo de datos long long pues no permitiría hacer los cálculos aritméticos correctamente, sobrepasando la mayor cantidad de dígitos que permitía almacenar este y cualquier otro tipo de dato primitivo.