

## 配列

### 利点

- ・サイズが固定されているからメモリ管理が楽。
- ・すごい軽量。
- ・連続したメモリ領域に格納されるから、インデックスによる要素アクセスがとても高速。
- ・.Max .Min で簡単に最大値と最小値を求めることができる。

### 欠点

- ・サイズが固定されているから、動的に要素を追加することや削除することが難しい。
- ・機能が少ない。基本的な要素の格納とアクセス機能くらいしかないから、要素の挿入、検索、ソートなどを手動で実装するしかない。

## ジェネリックコレクション

### 利点

- ・動的にサイズを変更できるから要素の追加や消去が簡単。
- ・要素の検索、ソート、並べ替え、フィルタリングなど機能が豊富。
- ・型安全性が高い。コンパイル時に型チェックを行うから実行時の型エラーを防ぐことができる。

### 欠点

- ・柔軟性と機能性が高いが故に配列と比べて多少のメモリオーバヘッド(余計な負荷)がある。
- ・要素アクセスや列挙のスピードが配列より若干遅い。
- ・少し複雑で扱いづらい

## 適している状況

### 配列

- ・サイズ変更の必要がなく、シンプルな処理にしか使用しない場合。
- ・データ量が少ない場合。(データが大きくなると処理が早いとはいえソートやフィルタリングが必要になりそう)
- ・処理速度を重視する場合。

### ジェネリックコレクション

- ・サイズ変更の必要があり、データ量が多くなる場合。