

```

ref class miki_ListeInt
{
private:
    Collections::Generic::List<int>^ liste;
public:
    /* Konstruktor */
    miki_ListeInt()
    {
        /* Erstellt liste aus einer Collections Klasse */
        liste=gcnew Collections::Generic::List<int>;
    }

    /* Setzt als Property das Wort Bearbeitung über das die
    verschiedenen Methoden der Collections Klasse aufgerufen werden
    können oder als Parameter auf das Listen Objekt zugegriffen werden */
    property Collections::Generic::List<int>^ Bearbeitung
    {
        Collections::Generic::List<int>^ get()
        {
            return(liste);
        }
    }

    /* Ausgabe des Arrays mit der For each schleife */
    void ausgabe()
    {
        for each(int i in liste)
        {
            Console::WriteLine(i);
        }
    }
    void sortAbwärts()
    {
        /* Sortiert algorithmus mit einer Soriter Klasse
        die eigentlich auf dem BubbleSort basiert */
        liste->Sort(gcnew IntAbwärtsVergleicher);
    }
    void sortAufwärts()
    {
        /* Sorier Methode ohne Parameter ruft automatisch
        die Aufwärtssoriter methode auf */
        liste->Sort();
    }
};

/* Sortier Klasse */
ref class IntAbwärtsVergleicher : Collections::Generic::IComparer<int> {
public:
    /* Bubbelsort durch virtualisierung der Methode Compare */
    virtual int Compare(int i1,int i2) {
        return(i2.CompareTo(i1));
    }
};

/* In Code */

miki::miki_ListeInt^ IntDyn = gcnew miki::miki_ListeInt();
IntDyn->Bearbeitung->Add(5);
IntDyn->Bearbeitung->Add(6);
IntDyn->ausgabe();|

```