

Sort

COLLABORATORS

	<i>TITLE :</i> Sort		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		January 23, 2025	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	Sort	1
1.1	Sort V1.00	1
1.2	sortdown	1
1.3	sortup	2

Chapter 1

Sort

1.1 Sort V1.00

PureBasic Sort library V1.00

Diese Library erlaubt Ihnen, numerische Arrays in aufsteigender oder absteigender Folge zu sortieren. Die benutzte Routine basiert auf dem QuickSort-Algorithmus, neu geschrieben in hoch-optimiertem Assembler. Dadurch ist sie möglicherweise eine der schnellsten Sortier-Routinen, die jemals für die 680x0 Prozessor-Familie verfügbar war.

Befehlsübersicht:

SortDown
SortUp

Beispiel:

Sort Demo

1.2 sortdown

SYNTAX

SortDown(array(), start.w, end.w)

STATEMENT

Sortiert das angegebene Array in absteigender Folge (große Zahlen zuerst).

array(): Name des zu sortierenden Arrays. Das Array muß ein Array vom Typ Byte, Word oder Long sein. String-Arrays werden nicht unterstützt.

start.w: Index für den Start des Sortierens. Sie können auch nur einen Teil des Arrays sortieren, falls notwendig.

end.w : Index für das Ende des Sortierens.

ACHTUNG: start.w und end.w müssen gültige Werte sein !

1.3 sortup

SYNTAX

```
SortUp(array(), start.w, end.w)
```

STATEMENT

Sortiert das angegebene Array in aufsteigender Folge (kleine Zahlen zuerst).

array(): Name des zu sortierenden Arrays. Das Array muß ein Array vom Typ Byte, Word oder Long sein. String-Arrays werden nicht unterstützt.

start.w: Index für den Start des Sortierens. Sie können auch nur einen Teil des Arrays sortieren, falls notwendig.

end.w : Index für das Ende des Sortierens.

ACHTUNG: start.w und end.w müssen gültige Werte sein !
