

ARBEITSBLATT ZU SORTIERVERFAHREN

Nachfolgend findest du ein Sortierverfahren in PROLOG implementiert, welches du in DELPHI bereits selbst implementiert hast (haben solltest!?). Das Prädikat `zliste` dient lediglich zu Testzwecken, um auf schnelle Art und Weise eine Liste von Zufallszahlen zu erhalten, welche sortiert werden soll.

```

/* Erzeugung einer Zufallsliste                                     */
/* N: Anzahl, G: Obergrenze, L: Liste                             */
zliste(0,G,[]).
zliste(N,G,[Z|Zs]):- N>0, N1 is N-1, random(0,G,Z), zliste(N1,G,Zs).

/* Hier ist das eigentliche Sortierverfahren                       */
sortiert([X],[X]).
sortiert(Xs,[X|Ys]):- kleinstes(Xs,X,Rest), sortiert(Rest,Ys).
kleinstes([X],X,[]).
kleinstes([K|Rs],K,Rs):- kleinstes(Rs,M,Hilfsrest), M>=K.
kleinstes([K|Rs],M,[K|Zs]):- kleinstes(Rs,M,Zs), M<K.

```

Ein Testlauf könnte somit wie folgt aussehen:

```

?- zliste(5,100,Ls), sortiert(Ls,Ss).
Ls = [69, 95, 2, 58, 64]
Ss = [2, 58, 64, 69, 95]

```

Aufgabe 1: Versuche an Hand des obigen Beispiels das Programm Schritt für Schritt nachzuvollziehen. Gib dem Sortierverfahren anschließend den aus Informatikersicht korrekten Namen.

Aufgabe 2: Erläutere die einzelnen Prädikate (`sortiert`, `kleinstes`) etwas genauer. Eine Kommentierung der Programmzeilen könnte dafür hilfreich sein.

Kommentiertes Programm zum _____-Sortierverfahren

```

sortiert([X],[X]).

sortiert(Xs,[X|Ys]):- kleinstes(Xs,X,Rest), sortiert(Rest,Ys).

kleinstes([X],X,[]).

kleinstes([K|Rs],K,Rs):- kleinstes(Rs,M,Hilfsrest), M>=K.

kleinstes([K|Rs],M,[K|Zs]):- kleinstes(Rs,M,Zs), M<K.

```

ARBEITSBLATT ZU SORTIERVERFAHREN

Nachfolgend findest du ein Sortierverfahren in PROLOG implementiert, welches du in DELPHI bereits selbst implementiert hast (haben solltest!?). Das Prädikat `zliste` dient lediglich zu Testzwecken, um auf schnelle Art und Weise eine Liste von Zufallszahlen zu erhalten, welche sortiert werden soll.

```

/* Erzeugung einer Zufallsliste                                     */
/* N: Anzahl, G: Obergrenze, L: Liste                             */
zliste(0,G,[]).
zliste(N,G,[Z|Zs]):- N>0, N1 is N-1, random(0,G,Z), zliste(N1,G,Zs).

/* Hier ist das eigentliche Sortierverfahren                       */
sortiert(Xs,Xs):- geordnet(Xs).
sortiert(Xs,Ys):- append(As,[X,Y|Rs],Xs), X>Y,
                  append(As,[Y,X|Rs],Xs1),sortiert(Xs1,Ys).
geordnet([_]).
geordnet([X,Y|Zs]):- X =< Y, geordnet([Y|Zs]).

```

Ein Testlauf könnte somit wie folgt aussehen:

```

?- zliste(5,100,Ls),sortiert(Ls,Ss).
Ls = [69, 95, 2, 58, 64]
Ss = [2, 58, 64, 69, 95]

```

Aufgabe 1: Versuche an Hand des obigen Beispiels das Programm Schritt für Schritt nachzuvollziehen. Gib dem Sortierverfahren anschließend den aus Informatikersicht korrekten Namen.

Aufgabe 2: Erläutere die einzelnen Prädikate (`sortiert`, `geordnet`) etwas genauer. Eine Kommentierung der Programmzeilen könnte dafür hilfreich sein.

Kommentiertes Programm zum _____-Sortierverfahren

```

sortiert(Xs,Xs):- geordnet(Xs).

sortiert(Xs,Ys):- append(As,[X,Y|Rs],Xs), X>Y,
                  append(As,[Y,X|Rs],Xs1),sortiert(Xs1,Ys).

geordnet([_]).

geordnet([X,Y|Zs]):- X =< Y, geordnet([Y|Zs]).

```

ARBEITSBLATT ZU SORTIERVERFAHREN

Nachfolgend findest du ein Sortierverfahren in PROLOG implementiert, welches du in DELPHI bereits selbst implementiert hast (haben solltest!?). Das Prädikat `zliste` dient lediglich zu Testzwecken, um auf schnelle Art und Weise eine Liste von Zufallszahlen zu erhalten, welche sortiert werden soll.

```

/* Erzeugung einer Zufallsliste                                     */
/* N: Anzahl, G: Obergrenze, L: Liste                             */
zliste(0,G,[]).
zliste(N,G,[Z|Zs]):- N>0, N1 is N-1, random(0,G,Z), zliste(N1,G,Zs).

/* Hier ist das eigentliche Sortierverfahren                     */
sortiert([],[]).
sortiert([X|Xs],Ys):- sortiert(Xs,Zs), eingefuegt(X,Zs,Ys).
eingefuegt(X,[],[X]).
eingefuegt(X,[Y|Ys],[X,Y|Ys]):- X =< Y.
eingefuegt(X,[Y|Ys],[Y|Zs]):- X>Y, eingefuegt(X,Ys,Zs).

```

Ein Testlauf könnte somit wie folgt aussehen:

```

?- zliste(5,100,Ls),sortiert(Ls,Ss).
Ls = [69, 95, 2, 58, 64]
Ss = [2, 58, 64, 69, 95]

```

Aufgabe 1: Versuche an Hand des obigen Beispiels das Programm Schritt für Schritt nachzuvollziehen. Gib dem Sortierverfahren anschließend den aus Informatikersicht korrekten Namen.

Aufgabe 2: Erläutere die einzelnen Prädikate (`sortiert`, `eingefuegt`) etwas genauer. Eine Kommentierung der Programmzeilen könnte dafür hilfreich sein.

Kommentiertes Programm zum _____-Sortierverfahren

```
sortiert([],[]).
```

```
sortiert([X|Xs],Ys):- sortiert(Xs,Zs), eingefuegt(X,Zs,Ys).
```

```
eingefuegt(X,[],[X]).
```

```
eingefuegt(X,[Y|Ys],[X,Y|Ys]):- X =< Y.
```

```
eingefuegt(X,[Y|Ys],[Y|Zs]):- X>Y, eingefuegt(X,Ys,Zs).
```

ZUSAMMENFASSUNG DER VERFAHREN

Kommentiertes Programm zum _____-Sortierverfahren

```
sortiert([X],[X]).
```

```
sortiert(Xs,[X|Ys]):- kleinstes(Xs,X,Rest),sortiert(Rest,Ys).
```

```
kleinstes([X],X,[]).
```

```
kleinstes([K|Rs],K,Rs):- kleinstes(Rs,M,Hilfsrest), M>=K.
```

```
kleinstes([K|Rs],M,[K|Zs]):- kleinstes(Rs,M,Zs), M<K.
```

Kommentiertes Programm zum _____-Sortierverfahren

```
sortiert(Xs,Xs):- geordnet(Xs).
```

```
sortiert(Xs,Ys):- append(As,[X,Y|Rs],Xs), X>Y,
```

```
append(As,[Y,X|Rs],Xs1),sortiert(Xs1,Ys).
```

```
geordnet([_]).
```

```
geordnet([X,Y|Zs]):- X =< Y, geordnet([Y|Zs]).
```

Kommentiertes Programm zum _____-Sortierverfahren

```
sortiert([],[]).
```

```
sortiert([X|Xs],Ys):- sortiert(Xs,Zs), eingefuegt(X,Zs,Ys).
```

```
eingefuegt(X,[],[X]).
```

```
eingefuegt(X,[Y|Ys],[X,Y|Ys]):- X =< Y.
```

```
eingefuegt(X,[Y|Ys],[Y|Zs]):- X>Y, eingefuegt(X,Ys,Zs).
```