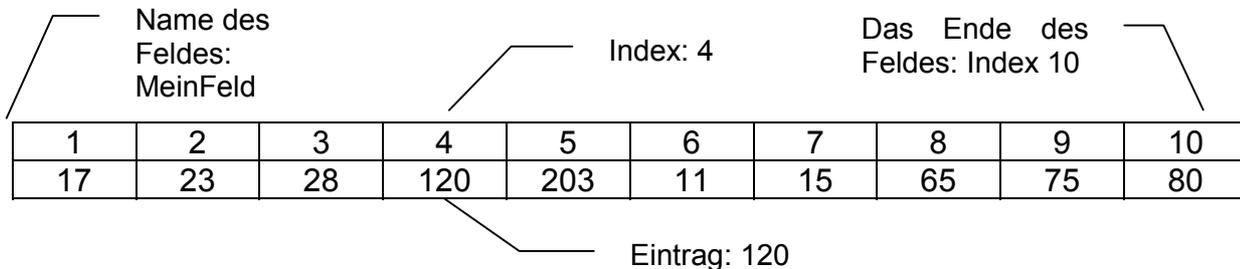


Durchlaufstrategien für eindimensionale Felder

Deklaration

```
VAR MeinFeld : ARRAY [1..10] of integer;
```



Durchlaufstrategien

Durchlaufstrategien ermöglichen das Durchlaufen eines Feldes um verschiedenster Aufgaben dabei zu erfüllen, z.B. Suchen, Selektieren, Sortieren usw. Wir unterscheiden drei elementare Strategien.

1. Bedingungsloses Durchlaufen:

Idee: Der Zeiger wird beim Ersten Feldelement angesetzt und dann wird das Feld schrittweise bis zum Ende durchlaufen.

Realisierung in Pascal: (For-To-Do-Schleife)

```
for zeiger := 1 to Ende do
begin
  {tue irgendetwas mit MeinFeld[zeiger] }
end;
```

2. Durchlaufen, bis eine Bedingung erfüllt ist:

Idee: Der Zeiger wird beim Ersten Feldelement angesetzt und dann wird das Feld schrittweise durchlaufen, bis eine bestimmte Bedingung erfüllt ist oder das Feldende erreicht ist.

Realisierung in Pascal: (Repeat-Until-Schleife)

```
zeiger:= 0;
repeat
  {tue irgendetwas mit MeinFeld[zeiger] }
  zeiger := zeiger +1;
until (Bedingung erfüllt) or (zeiger>=Ende);
```

3. Durchlaufen, solange eine Bedingung erfüllt ist:

Idee: Der Zeiger wird beim Ersten Feldelement angesetzt und dann wird das Feld schrittweise durchlaufen, solange eine bestimmte Bedingung erfüllt ist oder das Feldende noch nicht erreicht ist.

Realisierung in Pascal: (While-Do-Schleife)

```
zeiger:= 1;
while (Bedingung erfüllt) or (zeiger<=Ende)
do begin
  zeiger:=zeiger+1;
  {tue irgendetwas mit MeinFeld[zeiger] }
end;
```