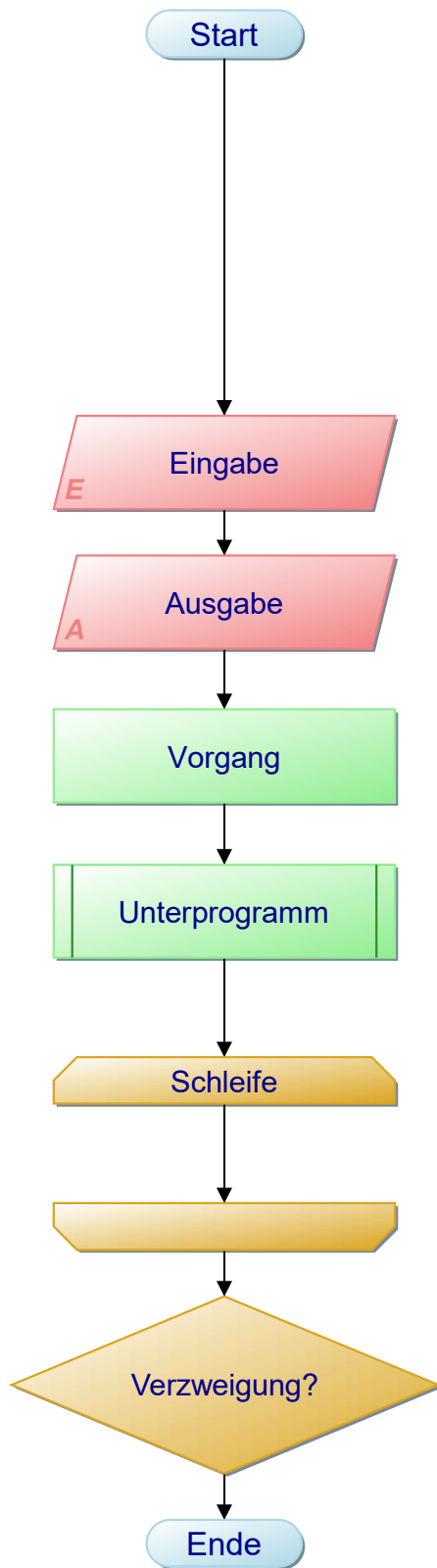


# Übersicht



Eine laufendes Programm stellt eine Abfolge von Vorgängen zur Lösung einer Aufgabe dar. Beim Programm sind es Programmierbefehle, mit deren Hilfe die Vorgänge ausgelöst werden.

Ein PAP kann ein Konstruktionsplan für ein Programm sein. Oder ein Programm kann als PAP dargestellt werden.

Unten sehen Sie, welche Sprachkonstrukte z.B. in der Sprache C# zu den PAP Symbolen passen.

`// Lesen von Konsole  
Console.ReadLine();`

`// Schreiben auf Konsole  
Console.WriteLine(....);`

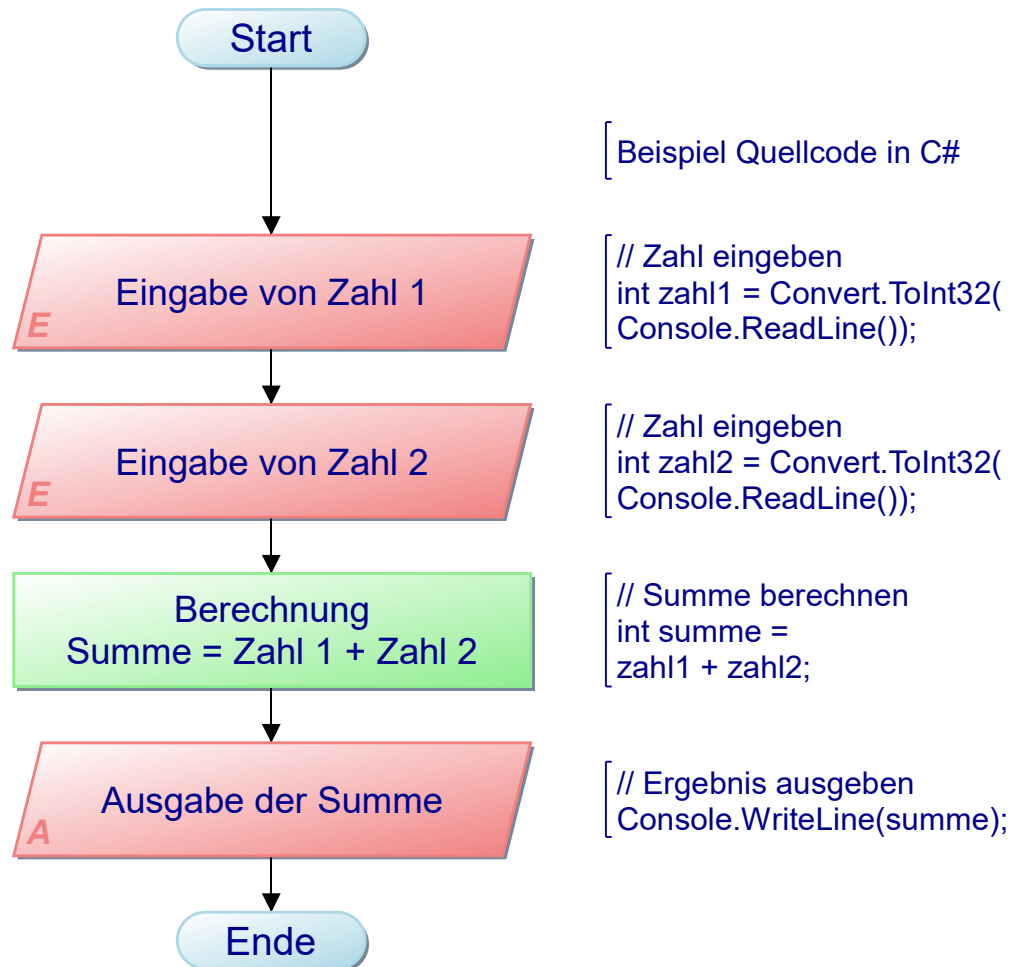
`// Gruppe von Anweisungen  
float y = m * x + b;  
string ys = "Ergebnis: " + y;  
Server.Send(ys);`

`// Methodenaufruf  
File.Sort("test.txt");`

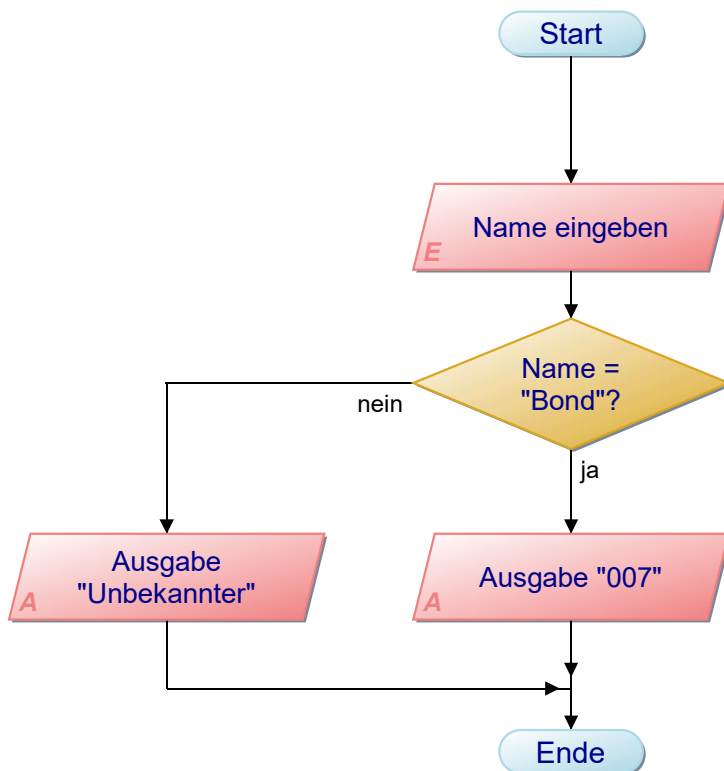
`// Die vier Schleifentypen  
for  
foreach  
while  
do...while`

`// Verzweigungstypen  
if  
if...else  
if...else if...else  
switch`

## Beispiel - Ein- und Ausgabe



## Beispiel - Verzweigung 1



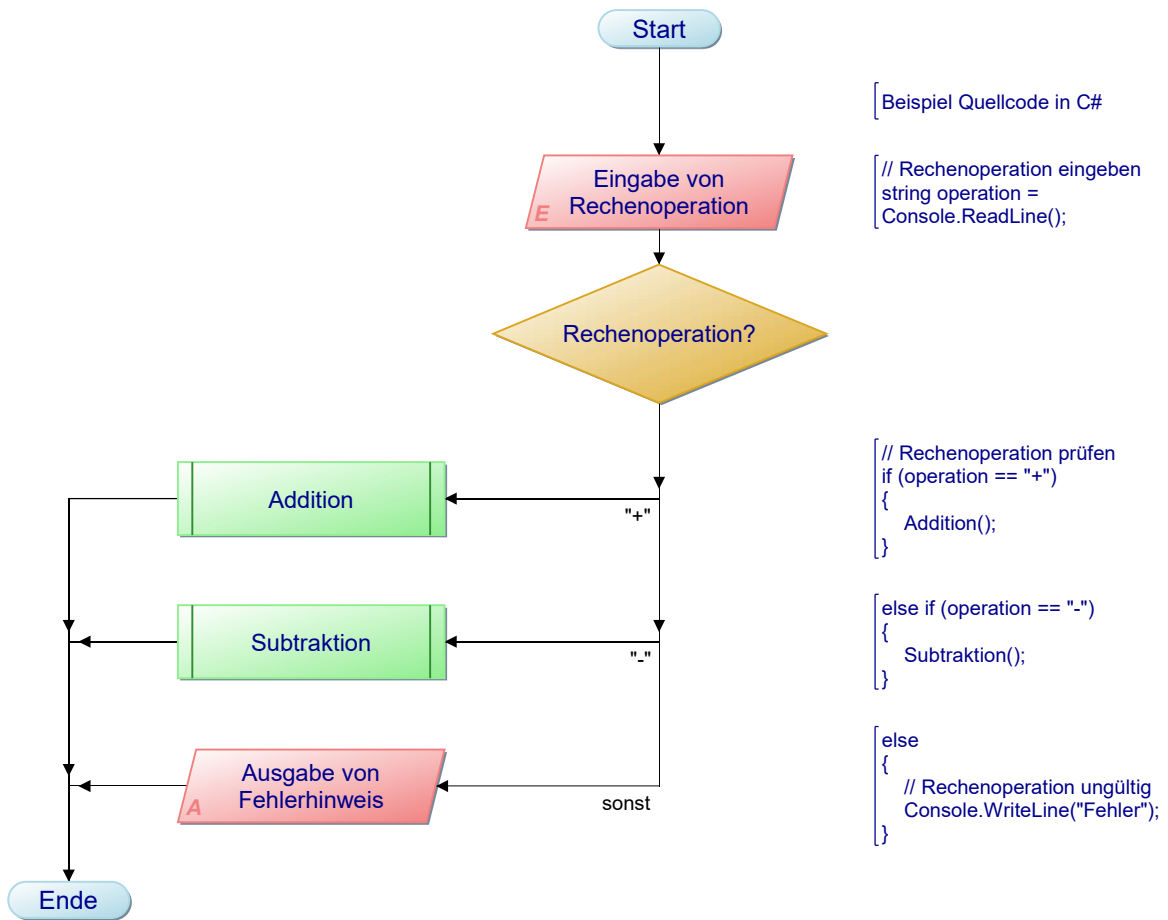
[ Beispiel Quellcode in C#

```
[ // Einen namen eingeben  
string name =  
Console.ReadLine();
```

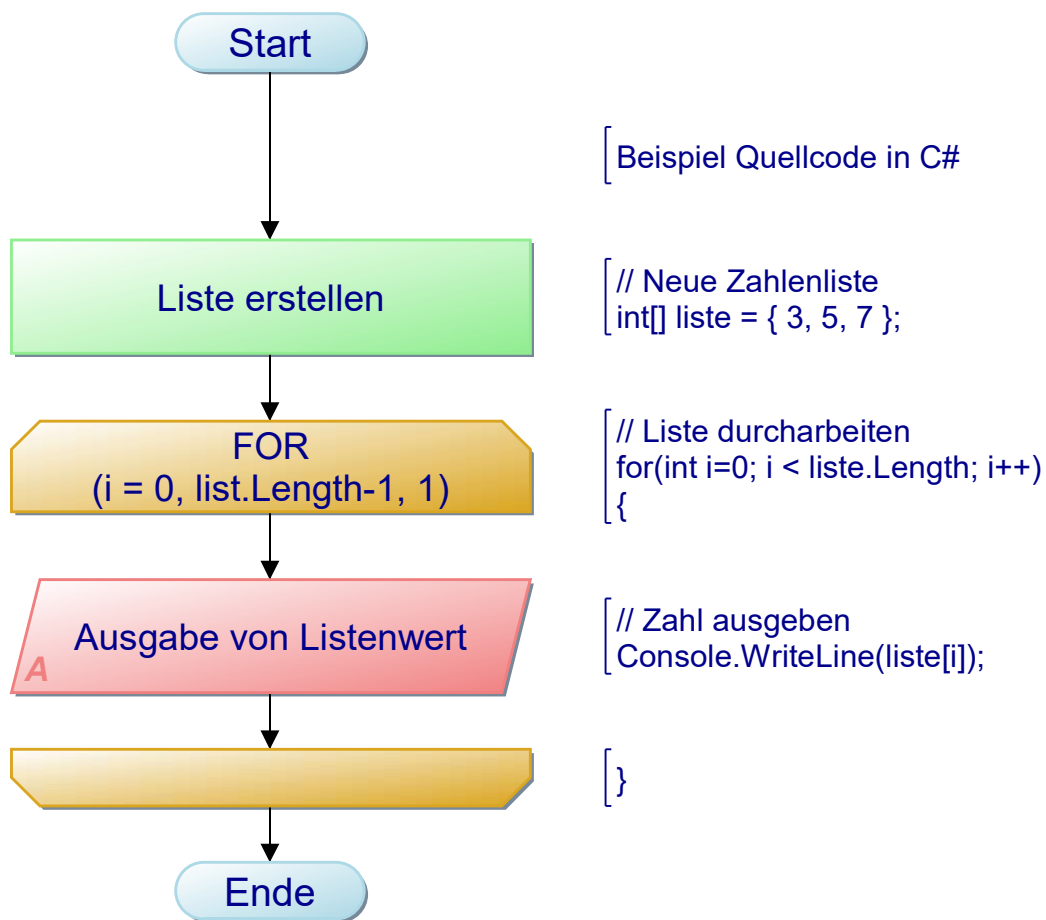
```
[ // Name überprüfen  
if (name == "Bond")  
{  
    Console.WriteLine("007");  
}
```

```
[ else  
{  
    Console.WriteLine(  
        "Unbekannter");  
}
```

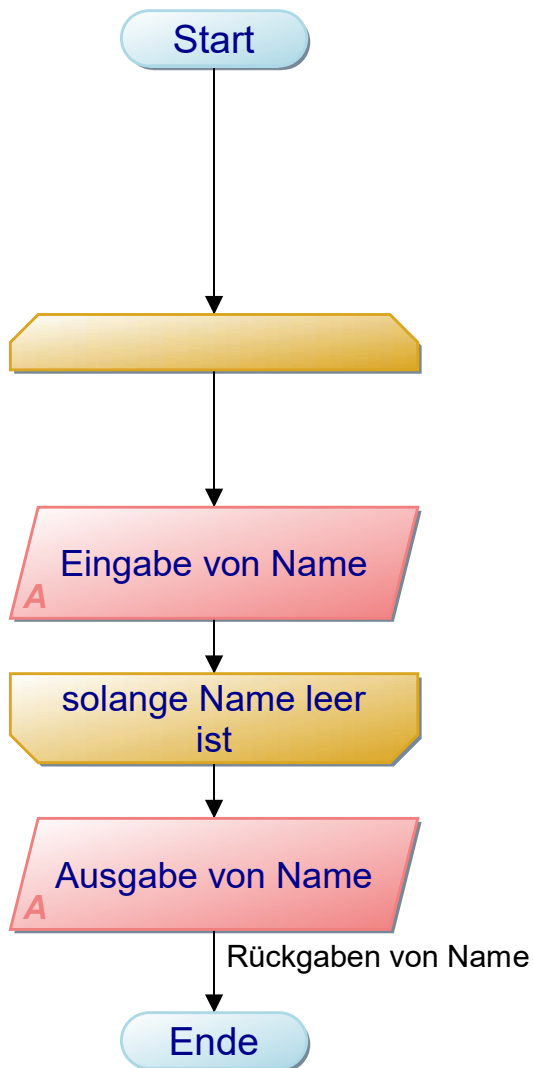
## Beispiel - Verzweigung 2



## Beispiel - For Schleife



## Beispiel - Do-While Schleife



[ Beispiel Quellcode in C#

```
[ // Arbeitsvariable außerhalb  
string name = "";  
  
// Ggf. wiederholen  
do  
{
```

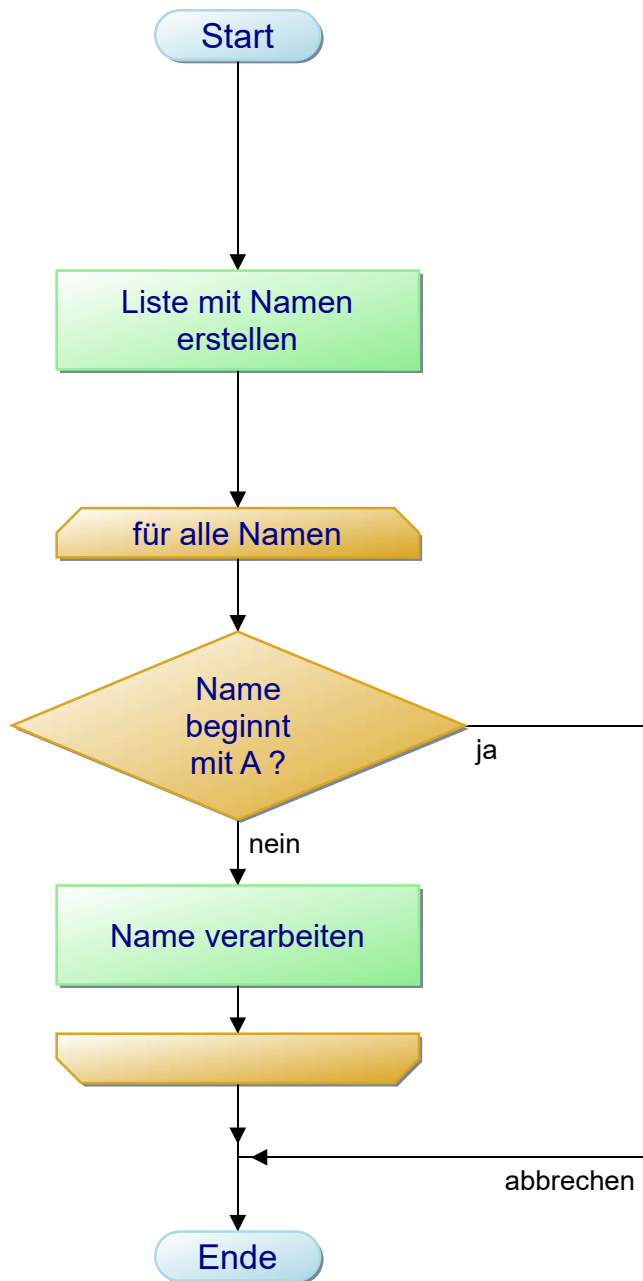
```
[ // Name eingeben  
name = Console.ReadLine();
```

```
[ // Eingabe prüfen  
} while (name == "");
```

```
[ // Bestätigung ausgeben  
Console.WriteLine(name);
```

```
[ // Name zurückgeben  
return name;
```

## Beispiel - Break



[ Beispiel Quellcode in C#

```
[ // Namenliste erzeugen  
int[] liste = {  
    "Hugo",  
    "Egon",  
    "Alfons",  
    "Willi",  
};
```

```
[ // Alle Namen durchgehen  
foreach(string name in liste)  
{
```

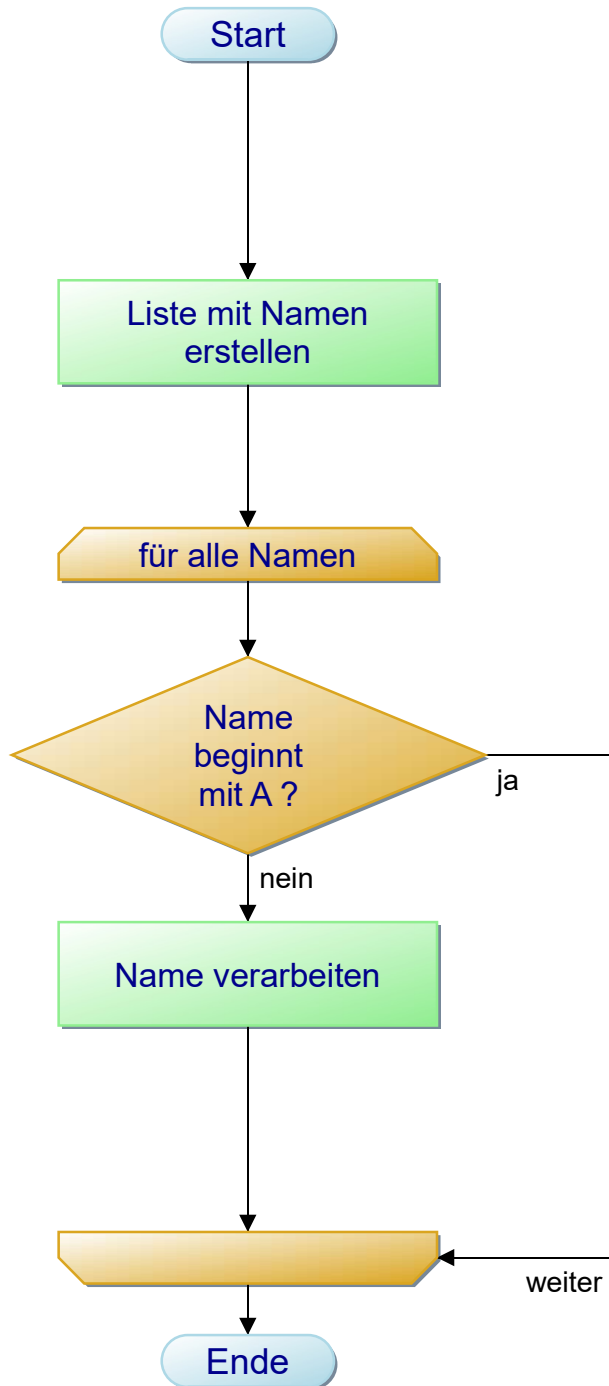
```
[ // Abbruchbedingung  
if (name.StartsWith("A"))  
{  
    break; // abbrechen  
}
```

```
[ // Listenwert bearbeiten  
NameVerarbeiten(name);
```

```
[ ]
```

```
[ // Schleife wird verlassen
```

## Beispiel - Continue



[ Beispiel Quellcode in C#

```
[ // Namenliste erzeugen  
int[] liste = {  
    "Hugo",  
    "Egon",  
    "Alfons",  
    "Willi",  
};
```

```
[ // Alle Namen durchgehen  
foreach(string name in liste)  
{
```

```
[ // Filterbedingung  
if (name.StartsWith("A"))  
{  
    continue; // überspringen  
}
```

```
[ // Listenwert bearbeiten  
NameVerarbeiten(name);
```

```
[ // überspringt Schritte bis  
Schleifenende und geht wieder zu  
Schleifenkopf
```

```
[  
}
```