

Freiwillige Offline-Aufgabe 07-04 (INF & WI & MCD):

Funktion `filter()`

(geübte C++ Konstrukte: struct, Arrays, Funktionen mit Array Parametern)

Gegeben sei die folgende `struct` zur Erfassung von Personendaten:

```
struct Person
{
    string nachname, vorname;
    int alter;
    char geschlecht;
};
```

Gegeben sei ferner das folgende Array zur Speicherung solcher Daten:

```
Person personen[100];
int anzahl_personen = 0;
```

Programmieren Sie zuerst ein Hauptprogramm, mit dem man Personendaten gemäß des folgenden Testlaufs einlesen kann:

Testlauf (Benutzereingaben sind unterstrichen):

```
Eine weitere Person eingeben (j/n)? j
Bitte den Nachnamen der 1. Person eingeben: ? Musterfrau
Bitte den Vornamen der 1. Person eingeben: ? Petra
Bitte das Alter der 1. Person eingeben: ? 33
Bitte das Geschlecht der 1. Person eingeben: ? w
Eine weitere Person eingeben (j/n)? j
Bitte den Nachnamen der 2. Person eingeben: ? Mustermann
Bitte den Vornamen der 2. Person eingeben: ? Peter
Bitte das Alter der 2. Person eingeben: ? 44
Bitte das Geschlecht der 2. Person eingeben: ? m
Eine weitere Person eingeben (j/n)? n
Drücken Sie eine beliebige Taste . . .
```

Das Hauptprogramm kann davon ausgehen, dass der Benutzer nur korrekte Eingaben macht, bis auf die `j/n` Eingabe: Diese soll explizit auf

Korrektheit geprüft werden, und bei Falscheingabe soll die Eingabeaufforderung wiederholt werden.

Testlauf (Benutzereingaben sind unterstrichen):

```
Eine weitere Person eingeben (j/n)? x
Eine weitere Person eingeben (j/n)? j
Bitte den Nachnamen der 1. Person eingeben: ? Musterfrau
Bitte den Vornamen der 1. Person eingeben: ? Petra
Bitte das Alter der 1. Person eingeben: ? 33
Bitte das Geschlecht der 1. Person eingeben: ? w
Eine weitere Person eingeben (j/n)? n
Drücken Sie eine beliebige Taste . . .
```

Sollten Sie in ihrem Hauptprogramm zwischen der Eingabe mittels `cin >>` und der Eingabe mittels `getline(...)` wechseln, so kann es notwendig werden, nach jeder Eingabe mittels `cin >> ein ...`

```
cin.clear();
cin.ignore();
```

... einzufügen, damit die Eingabe korrekt funktioniert.

Programmieren Sie nun eine Funktion

```
void filter( /* Parameter siehe Aufgabenstellung */ );
```

... welche das Personen-Array als ersten Parameter und `anzahl_personen` als zweiten Parameter entgegennimmt. Die Funktion soll vom Benutzer erfragen, welche Personendaten ausgegeben und gezählt werden sollen, und dies dann entsprechend umsetzen. Siehe Testläufe (der Teil ab „Ihre Auswahl“ werde durch die Funktion realisiert). Die Benutzereingabe braucht nicht auf Korrektheit geprüft zu werden.

Ihre Funktion soll aus dem Hauptprogramm aufgerufen werden. Ergänzen Sie dieses also entsprechend.

Um nicht beim Testen des Programms jedes Mal alle Personendaten neu eingeben zu müssen, ersetzen Sie bitte im Hauptprogramm die beiden Zeilen zur Definition des Arrays durch

```
Person personen[100] = {
{ "Musterfrau1", "Petra1", 18, 'w' },
{ "Mustermann1", "Klaus1", 18, 'm' },
{ "Mustermaedchen1", "Lisa1", 1, 'w' },
{ "Musterjunge1", "Jan1", 1, 'm' },
{ "Musterfrau2", "Petra2", 19, 'w' },
{ "Mustermann2", "Klaus2", 19, 'm' },
{ "Mustermaedchen2", "Lisa2", 2, 'w' },
{ "Musterjunge2", "Jan2", 2, 'm' },
{ "Musterfrau3", "Petra3", 20, 'w' },
{ "Mustermann3", "Klaus3", 20, 'm' },
{ "Mustermaedchen3", "Lisa3", 3, 'w' },
{ "Musterjunge3", "Jan3", 3, 'm' },
{ "Musterfrau4", "Petra4", 21, 'w' },
{ "Mustermann4", "Klaus4", 21, 'm' },
{ "Mustermaedchen4", "Lisa4", 4, 'w' },
{ "Musterjunge4", "Jan4", 4, 'm' },
{ "Musterfrau5", "Petra5", 22, 'w' },
{ "Mustermann5", "Klaus5", 22, 'm' },
{ "Mustermaedchen5", "Lisa5", 5, 'w' },
{ "Musterjunge5", "Jan5", 5, 'm' },
{ "Musterfrau6", "Petra6", 23, 'w' },
{ "Mustermann6", "Klaus6", 23, 'm' },
{ "Mustermaedchen6", "Lisa6", 6, 'w' },
{ "Musterjunge6", "Jan6", 6, 'm' },
{ "Musterfrau7", "Petra7", 24, 'w' },
{ "Mustermann7", "Klaus7", 24, 'm' },
{ "Mustermaedchen7", "Lisa7", 7, 'w' },
{ "Musterjunge7", "Jan7", 7, 'm' },
{ "Musterfrau8", "Petra8", 25, 'w' },
{ "Mustermann8", "Klaus8", 25, 'm' },
{ "Mustermaedchen8", "Lisa8", 8, 'w' },
{ "Musterjunge8", "Jan8", 8, 'm' }
};
int anzahl_personen = 32;
```

Testläufe des Gesamtprogramms, d.h. Hauptprogramm plus Funktion (Benutzereingaben sind unterstrichen):

```
Eine weitere Person eingeben (j/n)? n
Ihre Auswahl:
1 - Weibliche Erwachsene
2 - Maennliche Erwachsene
3 - Weibliche Kinder
4 - Maennliche Kinder
? 1
Musterfrau1, Petra1, w, 18
Musterfrau2, Petra2, w, 19
Musterfrau3, Petra3, w, 20
Musterfrau4, Petra4, w, 21
Musterfrau5, Petra5, w, 22
Musterfrau6, Petra6, w, 23
Musterfrau7, Petra7, w, 24
Musterfrau8, Petra8, w, 25
Summe: 8
Drücken Sie eine beliebige Taste . . .
```

```
Eine weitere Person eingeben (j/n)? n
Ihre Auswahl:
1 - Weibliche Erwachsene
2 - Maennliche Erwachsene
3 - Weibliche Kinder
4 - Maennliche Kinder
? 4
Musterjunge1, Jan1, m, 1
Musterjunge2, Jan2, m, 2
Musterjunge3, Jan3, m, 3
Musterjunge4, Jan4, m, 4
Musterjunge5, Jan5, m, 5
Musterjunge6, Jan6, m, 6
Musterjunge7, Jan7, m, 7
Musterjunge8, Jan8, m, 8
Summe: 8
Drücken Sie eine beliebige Taste . . .
```

Eine weitere Person eingeben (j/n)? j
Bitte den Nachnamen der 33. Person eingeben: ? Musterfrau33
Bitte den Vornamen der 33. Person eingeben: ? Petra
Bitte das Alter der 33. Person eingeben: ? 33
Bitte das Geschlecht der 33. Person eingeben: ? w
Eine weitere Person eingeben (j/n)? j
Bitte den Nachnamen der 34. Person eingeben: ? Mustermann44
Bitte den Vornamen der 34. Person eingeben: ? Peter
Bitte das Alter der 34. Person eingeben: ? 44
Bitte das Geschlecht der 34. Person eingeben: ? m
Eine weitere Person eingeben (j/n)? n
Ihre Auswahl:
1 - Weibliche Erwachsene
2 - Maennliche Erwachsene
3 - Weibliche Kinder
4 - Maennliche Kinder
? 1
Musterfrau1, Petra1, w, 18
Musterfrau2, Petra2, w, 19
Musterfrau3, Petra3, w, 20
Musterfrau4, Petra4, w, 21
Musterfrau5, Petra5, w, 22
Musterfrau6, Petra6, w, 23
Musterfrau7, Petra7, w, 24
Musterfrau8, Petra8, w, 25
Musterfrau33, Petra, w, 33
Summe: 9
Drücken Sie eine beliebige Taste . . .
