

```

#include <iostream>
#include <iomanip>
using namespace std;

/* Struktur eines Listenelements */
struct FloatListElem {
    float data;
    FloatListElem *pNext;
};

/* Liste ausgeben mit gegebener Genauigkeit */
void fl_print(FloatListElem * const pAnchor, int precision=2)
{
    // Hilfszeiger auf den Anker (Beginn) der Liste
    FloatListElem *pHelp = pAnchor;
    int i=0;
    // Anzahl Nachkommastellen festlegen
    cout.precision(precision);
    // Liste bis zum Ende durchlaufen
    while(pHelp != NULL)
    {
        // nach 3 Zahlen einen Zeilenumbruch
        if(i%3 == 0 && i>0)
            cout << endl;
        // Darstellung im 15 Zeichen breiten Spalten
        // und in Gleitpunktdarstellung
        cout << setw(15) << fixed << pHelp->data;
        // naechstes Element betrachten
        pHelp = pHelp->pNext;
        i++;
    }
    if(i==0)
        cout << "Noch keine Elemente in der Liste!";
    cout << endl;
}

```

```

/* komplette Liste leeren */
void fl_delete(FloatListElem *& pAnchor)
{
    // Hilfszeiger auf den Anker (Beginn) der Liste
    FloatListElem *pHelp = pAnchor;
    // Liste bis zum Ende durchlaufen
    while(pHelp != NULL){
        // Zeiger auf zu löschendes Element definieren
        FloatListElem *pDelete = pHelp;
        // naechstes Element betrachten
        pHelp = pHelp->pNext;
        // altes Element loeschen
        delete pDelete;
    }
    // Anker zurueck auf NULL setzen (leere Liste)
    pAnchor = NULL;
}

/* Hauptprogrammfunktion */
int main()
{
    // Zeiger auf FloatListElem definieren
    FloatListElem *fl = NULL;
    // Liste mit Genauigkeit 2 ausgeben
    fl_print(fl,2);
    // Liste freigeben
    fl_delete(fl);
}

```