

Float-Vektor-Funktionen – Speicherskizzen

Im Folgenden sollen die von der **main()-Funktion** aufgerufenen Befehle bildhaft verdeutlicht werden. Dabei sind die Speicherzellen jeweils zu ergänzen.

FloatVec fv;

- Erzeugung eines Objektes namens „fv“ mit dem Datentyp „FloatVec“

	Mitgliedername	Adresse	Inhalt
<u>Objekt fv:</u>	vecPtr	0xbffff110	
	maxmem	0xbffff114	
	used	0xbffff118	

fv_init(fv, 5);

- in dieser Funktion wird v wird zu einer Referenz (Aliasname) für fv und maxmem mit 5 initialisiert

```
v.vecPtr = new float[maxmem];
```

- neues Speicherfeld der Grösse maxmem erzeugen und den Anfang der Adresse des neuen Speichers an v.vecPtr übergeben

```
v.maxmem = maxmem;
```

```
v.used = 0;
```

- Anzahl max. verfügbarer Vektor-Elemente auf maxmem und Anzahl benutzter Elemente auf 0 setzen

Objekt fv:

Mitgliedername	Adresse	Inhalt
vecPtr	0xbffff110	
maxmem	0xbffff114	
used	0xbffff118	

Speicherfeld:

	Adresse	Inhalt
vecPtr[0]	0x804a008	
vecPtr[1]	0x804a00c	
vecPtr[2]	0x804a010	
vecPtr[3]	0x804a014	
vecPtr[4]	0x804a018	

```
for(float i=0.1; i<2.0; i+=0.4)
    fv_append(fv,i);
```

Objekt fv:

Mitgliedername	Adresse	Inhalt
vecPtr	0xbffff110	
maxmem	0xbffff114	
used	0xbffff118	

Speicherfeld:

	Adresse	Inhalt
vecPtr[0]	0x804a008	
vecPtr[1]	0x804a00c	
vecPtr[2]	0x804a010	
vecPtr[3]	0x804a014	
vecPtr[4]	0x804a018	

fv_print(fv, 5);

- v wird wieder zur Referenz für fv
- Ausgabe des Float-Vektors v mit gewünschter Anzahl Nachkommastellen

fv_append(fv, 123.45);

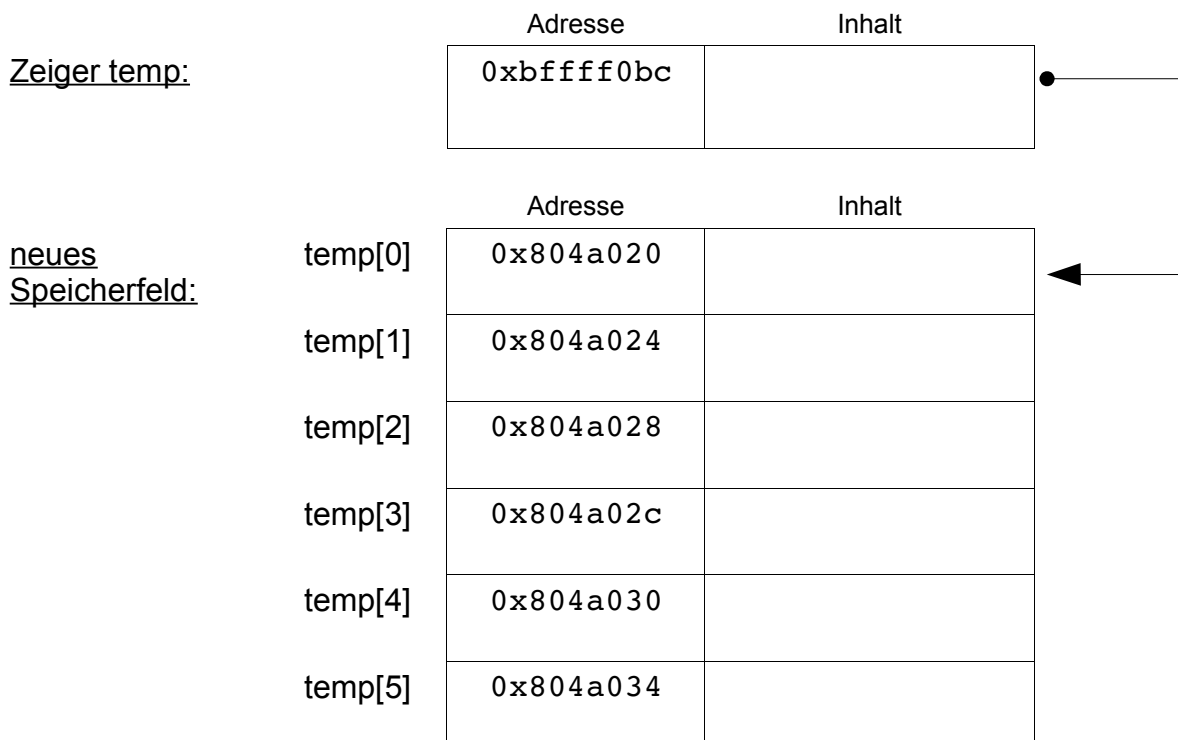
- v wird wieder zur Referenz für fv
- nun ist der maximal verfügbare Speicher ausgereizt: `v.used == v.maxmem`

fv_resize(v, v.maxmem+1);

- um aktuelles Speicherfeld zu vergrößern, muss ein neues erzeugt, der Inhalt des alten Feldes in das neue Feld kopiert und das alte Feld gelöscht werden

`float *temp = new float[size];`

- neues Speicherfeld der Größe size erzeugen und den Anfang der Adresse des neuen Speichers an temp übergeben

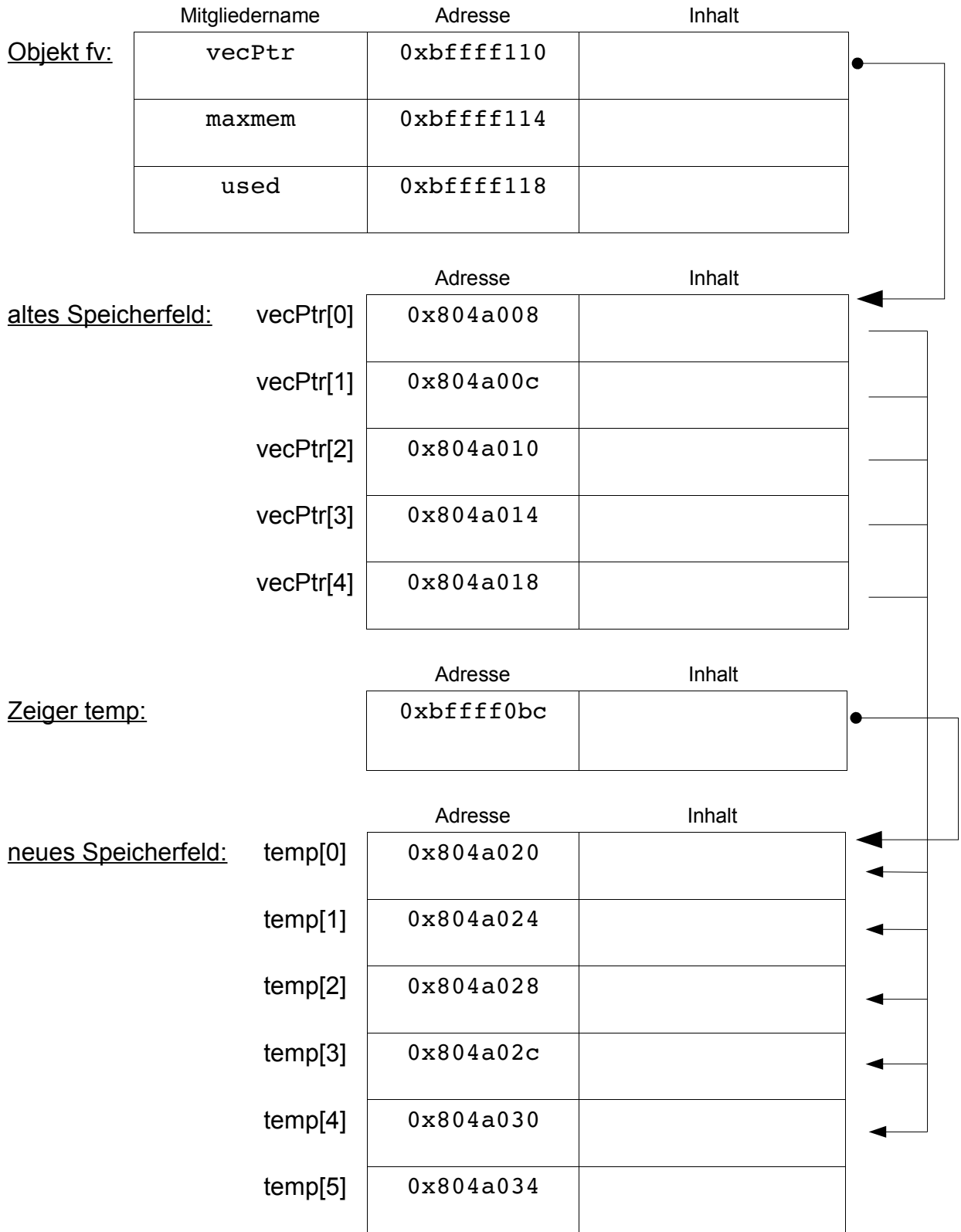


```

for(int i=0; i<v.used; i++)
    *(temp+i) = v.vecPtr[i];

```

- Werte vom alten Feld ins neue Feld kopieren

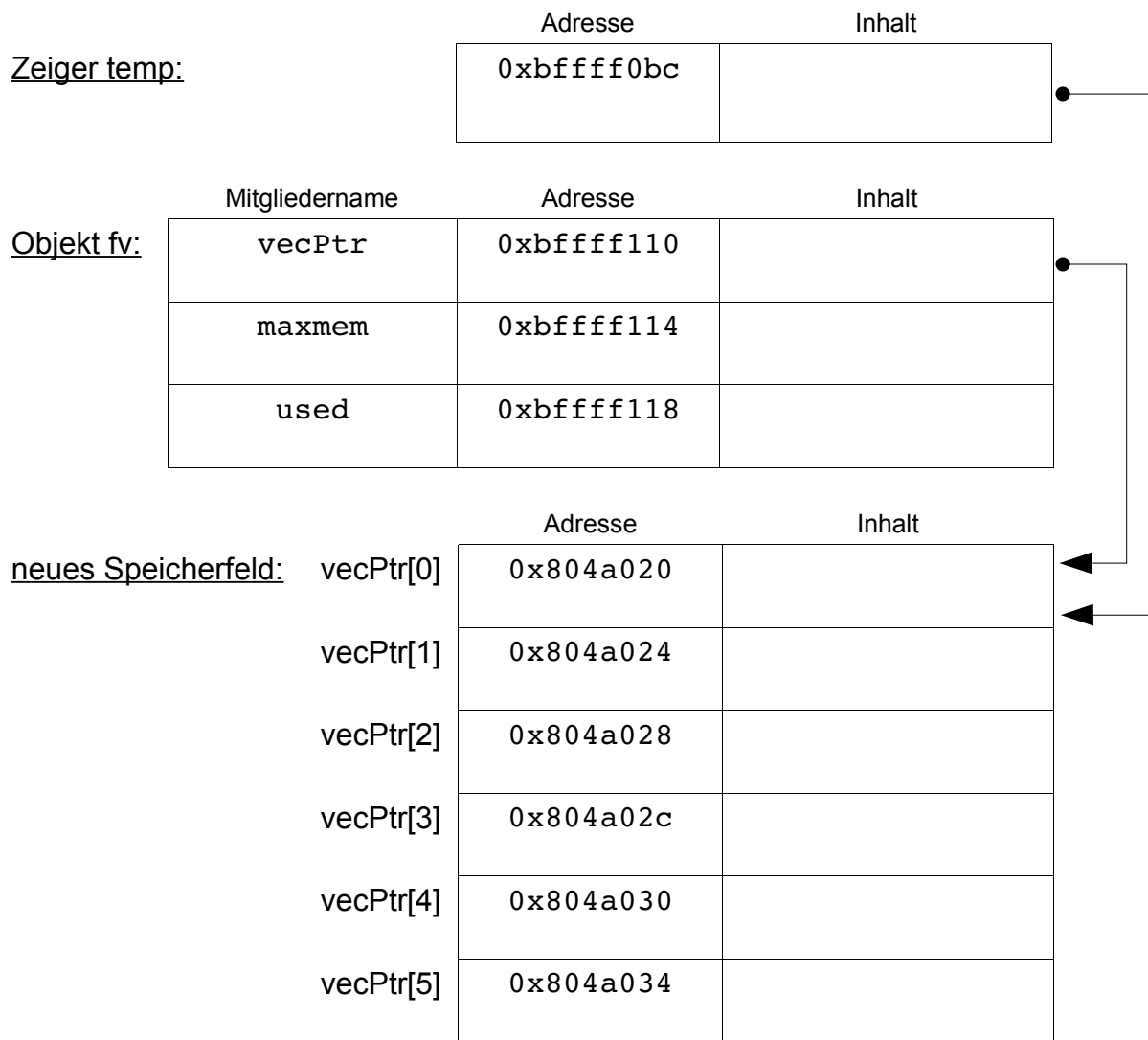


delete[] v.vecPtr;

- Löschen der Speicherzellen des alten Feldspeichers:
0x804a008 bis 0x804a018
- v.vecPtr zeigt nun auf NULL

v.vecPtr = temp;

- den Zeiger vecPtr auf den neuen Speicherplatz zeigen lassen, auf den bisher nur temp zeigte (temp verschwindet nach Funktionsende)



fv_print(fv, 2);

- v wird wieder zur Referenz für fv
- Ausgabe des Float-Vektors v mit gewünschter Anzahl Nachkommastellen

fv_delete(fv);

- vor dem Programmende räumt diese Funktion den Speicher auf und gibt den dynamischen Speicher des Vektor-Feldes frei
- es bleibt wie nach FloatVec fv; nur noch das Objekt fv im Speicher

	Mitgliedername	Adresse	Inhalt
<u>Objekt fv:</u>	vecPtr	0xbffff110	
	maxmem	0xbffff114	
	used	0xbffff118	