

```

1  #include <iostream>
2  using std::cout;
3  using std::endl;
4
5  class CDemo
6  {
7      int iNo;
8      float fNo;
9      double dNo;
10     char chChar;
11
12     int *pPtr1;
13     int *pPtr2;
14
15 public:
16     CDemo()
17     {
18         iNo = 0;
19         fNo = 0.0f;
20         dNo = 0.0;
21         chChar = '\0';
22
23         pPtr1 = NULL;
24
25         pPtr2 = new int[3];
26         if(NULL == pPtr2)
27         {
28             cout << "Memory allocation FAILED";
29             return;
30         }
31
32         /*
33         for(int iCounter = 0; iCounter < 3; iCounter++)
34             pPtr2[iCounter] = 0;
35         */
36
37         memset(pPtr2, 0, 3 * sizeof(int));
38     }
39
40     void Set(int iParam, float fParam, double dParam, char chParam)
41     {
42         iNo = iParam;
43         fNo = fParam;
44         dNo = dParam;
45         chChar = chParam;
46
47         pPtr1 = &iNo;
48
49         for(int iCounter = 0; iCounter < 3; iCounter++)
50             pPtr2[iCounter] = iParam + iCounter + 1;
51     }
52
53     void Get()
54     {
55         cout << "iNo is " << iNo << endl;
56         cout << "fNo is " << fNo << endl;
57         cout << "dNo is " << dNo << endl;
58         cout << "chChar is " << chChar << endl;
59
60         cout << "&iNo is " << &iNo << endl;
61         cout << "pPtr1 is " << pPtr1 << endl;
62         if(pPtr1 != NULL)
63             cout << "**pPtr1 is " << *pPtr1 << endl;
64
65         cout << "pPtr2 is " << pPtr2 << endl;
66         cout << "pPtr2 values are:\t";
67         for(int iCounter = 0; iCounter < 3; iCounter++)

```

```

68         cout << "pPtr2[" << iCounter << "] = " << pPtr2[iCounter] << "\t";
69
70     cout << endl;
71 }
72
73 CDemo(CDemo &refObj)
74 {
75     cout << "\nIn copy constructor\n";
76
77     iNo = refObj.iNo;
78     fNo = refObj.fNo;
79     dNo = refObj.dNo;
80     chChar = refObj.chChar;
81
82     pPtr1 = refObj.pPtr1;
83     pPtr2 = refObj.pPtr2;
84 }
85
86 CDemo& operator =(CDemo &refOperand2)
87 {
88     cout << "\nIn assignment operator\n";
89
90     iNo = refOperand2.iNo;
91     fNo = refOperand2.fNo;
92     dNo = refOperand2.dNo;
93     chChar = refOperand2.chChar;
94
95     pPtr1 = refOperand2.pPtr1;
96     pPtr2 = refOperand2.pPtr2; // old memory leaked
97
98     return *this;
99 }
100
101 ~CDemo()
102 {
103     iNo = 0;
104     fNo = 0.0f;
105     dNo = 0.0;
106     chChar = '\0';
107
108     pPtr1 = NULL;
109
110     if(pPtr2 != NULL)
111     {
112         delete pPtr2; // there will be crash if another objects access memory
113         pPtr2 = NULL;
114     }
115 }
116 };
117
118 int main(void)
119 {
120     CDemo obj1;
121
122     cout << "OBJ1 values are:\n";
123     obj1.Get();
124
125     obj1.Set(10, 57.33f, 69.33, 'A');
126     cout << "\nOBJ1 values are:\n";
127     obj1.Get();
128
129     CDemo obj2 = obj1;
130     cout << "\nOBJ2 values are:\n";
131     obj2.Get();
132
133     CDemo obj3;
134     obj3 = obj1;

```

```
135     cout << "\nOBJ3 values are:\n";
136     obj3.Get();
137
138     return 0;
139 }
140
```