

```

1  #include <iostream>
2  #include <iomanip>
3  using std::cout;
4  using std::endl;
5  using std::setw;
6
7  class demo
8  {
9      char chChar;
10     int iNo;
11     float fNo;
12     double dNo;
13
14     int *pPtr1;
15     int *pPtr2;
16
17 public:
18     demo()
19     {
20         chChar = '\0';
21         iNo = 0;
22         fNo = 0.0f;
23         dNo = 0.0;
24
25         pPtr1 = NULL;
26
27         pPtr2 = new int[3];
28         if(NULL == pPtr2)
29         {
30             cout << "Memory allocation FAILED\n";
31             return;
32         }
33
34         /*
35         for(int iCounter = 0; iCounter < 3; iCounter++)
36             pPtr2[iCounter] = 0;
37         */
38
39         memset(pPtr2, 0, 3 * sizeof(int));
40     }
41
42     ~demo()
43     {
44         chChar = '\0';
45         iNo = 0;
46         fNo = 0.0f;
47         dNo = 0.0;
48
49         pPtr1 = NULL;
50
51         if(pPtr2 != NULL)
52         {
53             delete []pPtr2;
54             pPtr2 = NULL;
55         }
56     }
57
58     void set(char chParam, int iParam, float fParam, double dParam)
59     {
60         chChar = chParam;
61         iNo = iParam;
62         fNo = fParam;
63         dNo = dParam;
64
65         pPtr1 = &iNo;
66
67         if(pPtr2 != NULL)

```

```

68         {
69             for(int iCounter = 0; iCounter < 3; iCounter++)
70                 pPtr2[iCounter] = iParam + iCounter + 1;
71         }
72     }
73
74     void get()
75     {
76         cout << "Character is " << chChar << endl;
77         cout << "Integer is " << iNo << endl;
78         cout << "Float is " << fNo << endl;
79         cout << "Double is " << dNo << endl;
80
81         cout << "&iNo is " << &iNo << endl;
82         cout << "pPtr1 is " << pPtr1 << endl;
83         if(pPtr1 != NULL)
84             cout << "*pPtr1 is " << *pPtr1 << endl;
85
86         cout << "pPtr2 is " << pPtr2 << endl;
87         cout << "pPtr2 values are :\n";
88         if(pPtr2 != NULL)
89             {
90                 for(int iCounter = 0; iCounter < 3; iCounter++)
91                     cout << pPtr2[iCounter] << setw(20);
92                 cout << endl;
93             }
94     }
95
96     demo(demo &refobj)
97     {
98         cout << "\nIn copy constructor\n";
99         chChar = refobj.chChar;
100        iNo = refobj.iNo;
101        fNo = refobj.fNo;
102        dNo = refobj.dNo;
103
104        pPtr1 = &iNo;
105        pPtr2 = new int[3];
106        if(NULL == pPtr2)
107        {
108            cout << "Memory allocation FAILED\n";
109            return;
110        }
111
112        /*
113        for(int iCounter = 0; iCounter < 3; iCounter++)
114            pPtr2[iCounter] = refobj.pPtr2[iCounter];
115        */
116
117        memcpy(pPtr2, refobj.pPtr2, 3 * sizeof(int));
118    }
119
120     demo& operator =(demo &refobj2)
121     {
122         cout << "\nIn assignment operator\n";
123         chChar = refobj2.chChar;
124         iNo = refobj2.iNo;
125         fNo = refobj2.fNo;
126         dNo = refobj2.dNo;
127
128         if(refobj2.pPtr1 != NULL)
129             pPtr1 = &iNo;
130
131         // removed memory allocation code as already allocated in constructor
132
133         /*
134         for(int iCounter = 0; iCounter < 3; iCounter++)

```

```

135         pPtr2[iCounter] = refobj.pPtr2[iCounter];
136     */
137
138     if(pPtr2 != NULL)
139         memcpy(pPtr2, refobj2.pPtr2, 3 * sizeof(int));
140
141     return *this;
142 }
143 };
144
145 int main(void)
146 {
147     demo obj1;
148     cout << "OBJ1:\n";
149     obj1.get();
150
151     obj1.set('A', 10, 57.33f, 69.33);
152     cout << "\nOBJ1:\n";
153     obj1.get();
154
155     demo obj2 = obj1;
156     cout << "\nOBJ2:\n";
157     obj2.get();
158
159     demo obj3;
160     obj3 = obj1;
161     cout << "\nOBJ3:\n";
162     obj3.get();
163
164     cout << "Done\n";
165
166     return 0;
167 }
168

```