

```

1  #include "postac.h"
2  int main()
3  {
4      Obiekt* o1 = new Postac('P');
5      Obiekt* o2 = new Przedmiot('#', "mur");
6
7      Plansza p(3,5);
8
9      cout<<p<<endl;
10
11     p.dodaj(0,0,o1);
12     p.dodaj(2,4,o2);
13
14     cout<<p<<endl;
15
16     p-='#';
17
18     cout<<p<<endl;
19
20     return 0;
21 }
22
23 #ifndef ZPO
24 #define ZPO
25
26 #include <iostream>
27 using namespace std;
28
29 class Obiekt
30 {
31 private:
32     char znak;
33 public:
34     Obiekt(char znak): znak(znak) {}
35     virtual ~Obiekt()=default;
36     virtual void print() = 0;
37     char getznak() {return znak;}
38 };
39
40 class Postac : public Obiekt
41 {
42 public:
43     Postac(char znak) : Obiekt(znak) {}
44     ~Postac()
45     {
46         cout<<"\n("<<getznak()<<") deleted"<<endl;
47     }
48
49     void print() override
50     {

```

```

51         cout<<getznak()<<" ";
52     }
53 };
54
55 class Przedmiot : public Obiekt
56 {
57 private:
58     string nazwa;
59 public:
60     Przedmiot(char znak, string nazwa) : Obiekt(znak),
nazwa(nazwa) {}
61     ~Przedmiot()
62     {
63         cout<<endl<<nazwa<<" ("<<getznak()<<" ) deleted"<<endl;
64     }
65     void print() override
66     {
67         cout<<getznak()<<" ";
68     }
69 };
70
71 class Plansza
72 {
73 private:
74     int x;
75     int y;
76     Obiekt* **tab;
77 public:
78     Plansza(int x, int y) : x(x), y(y)
79     {
80         tab = new Obiekt* *[x];
81         for(int i=0; i<x;i++)
82         {
83
84             tab[i]=new Obiekt*[y];
85             for(int j=0;j<y;j++)
86             {
87                 tab[i][j]=nullptr;
88             }
89         }
90     }
91     ~Plansza()
92     {
93         for(int i=0;i<x;i++)
94         {
95             for(int j=0;j<y;j++)
96             {
97                 if(tab[i][j]!=nullptr) delete tab[i][j];
98             }
99             delete []tab[i];

```

```

100         }
101         delete []tab;
102
103     }
104
105     void dodaj(int x, int y, Obiekt* o)
106     {
107         tab[x][y]=o;
108     }
109
110     friend ostream& operator<<(ostream& os, const Plansza& pl)
111     {
112         for(int i=0; i<pl.x; i++)
113         {
114             for(int j=0; j<pl.y; j++)
115             {
116                 if(pl.tab[i][j]!=nullptr)
117                 {
118                     pl.tab[i][j]->print();
119                 }
120                 else os<<" ";
121             }
122             os<<endl;
123         }
124         return os;
125     }
126
127     void operator--(char znak)
128     {
129         for(int i=0; i<x; i++)
130         {
131             for(int j=0; j<y; j++)
132             {
133                 if(tab[i][j]!=nullptr &&
134                 tab[i][j]->getznak()==znak)
135                 {
136                     delete tab[i][j];
137                     tab[i][j]=nullptr;
138                 }
139             }
140         }
141
142     };
143
144 #endif // ZPO
145

```