```
#include <iostream>
void MinStack::push(int d)
{
Node* t = new Node(d);
if(top == NULL)
{
top = min = t;
}
else {
t->link = top;
top = t;
if(min->data > top->data)
{
top->nextMin = min;
min = top;
}
}}
int MinStack::pop()
{
if(top == NULL) return -1;
if(min == top)
{
min = min->nextMin;
}
int d = top->data;
Node* t = top;
top = top->link;
delete t;
return d;
}
```

```
delete t;
return d;
}
int MinStack::getMinimum()
{
if(min == NULL)
{
return -1;
}
 else {
 return min->data;
 }
 }
 void MinStack::printStack()
 {
 for (Node*t = top; t!=NULL; t=t->link)
 {
 cout << " " << t->data;
 }
 }
```