

```
1  #include <iostream>
2
3  using namespace std;
4
5  class Stack
6  {
7  public:
8  Stack();
9  ~Stack();
10 void push(int);
11 int pop();
12 int peek();
13 friend void print(Stack&); 14
15 private:
```

تعريف العقدة:

```
16 typedef struct node {
17     node *next;
18     int data;
19 } NODE;
20
21 NODE *top;
22 };
23
```

التابع البناني والهادم:

```
24 Stack::Stack()
25 {
26     top = NULL;
27 }
28
29 Stack::~Stack()
30 {
```

```
31 while(top) {
32     NODE *tmp = top;
33     top = top->next;
34     delete tmp;
35 }
36 }
37
```

تابع اضافة عنصر الى المكس:

```
38 void Stack::push(int n) 39 {
40     NODE *ptr = new NODE;
41     ptr->next = top;
42     ptr->data = n;
43     top = ptr; 44 }
45
```

تابع نزع عنصر من المكس:

```
46 int Stack::pop()
47 {
48     NODE *tmp = top;
49     int val = top->data;
50     top = top->next;
51     delete tmp;
52     return val;
53 }
54
```

تابع يعيد العنصر في قمة المكس:

```
55 int Stack::peek()
56 {
57     return top->data;
58 }
59
```

تابع طباعة محتويات مكس:

```
60 void print(Stack &s)
61 {
```



```
62  Stack::NODE *cur = s.top;
63  while(cur) {
64  cout << cur->data << " ";
65  cur = cur->next;
66  }
67  cout << endl;
68  }
69
70  int main()
71  {
72  Stack *st = new Stack;
73  st->push(10);
74  st->push(20);
75  st->push(30);
76  st->push(40);
77  st->push(50);
78  print(*st);
79  st->pop();
80  st->pop();
81  print(*st);
82  cout << "current top=" << st->peek();
83  return 0;
84  }
```

التابع الرئيسي: