

```
1  /*File : stack01.cpp
2      deklarasi stack yang diimplementasi dengan tabel kontigu
3      dan ukuran sama
4      Top adalah alamat elemen puncak, Top dimulai dari index 1 - MaxEl+1
5      implementasi dalam bahasa C++ dengan aloksi statik
6
7      Author : G. Ngr Mega Nata
8      tanggal: 19 Maret 2016
9  */
10
11  #include<iostream.h>
12  #include<conio.h>
13  #include<stdio.h>
14  #include<stdlib.h>
15  #define MaxEl 5
16
17  typedef int infotype;
18  typedef int address;
19
20  //Top dimulai dari index 1 - MaxEl+1
21  struct Stack{
22      infotype T[MaxEl+1];
23      address Top;
24  };
25
26  #define Top(S) (S).Top
27  #define InfoTop(S) (S).T[(S).Top]
28
29  //Prototype
30  void CreateEmpty(Stack *S);
31  bool IsEmpty (Stack S);
32  bool IsFull(Stack S);
33  void Push(Stack *S,infotype x);
34  void Pop(Stack *S,infotype *x);
35  void Cetak(Stack S);
36
37  int main(){
38      Stack S;
39      int x;
40
41      CreateEmpty(&S);
42      Pop(&S,&x); //pop pada stack kosong
43
44      Push(&S,3);   Push(&S,6);
45      Push(&S,7);   Push(&S,9);
46      Push(&S,12);
47
48      Push(&S,2); //push pada stack full
49
50      Pop(&S,&x);
51      Pop(&S,&x);
52
53      Cetak(S);
54
55      getch();
56  return 0;
57  }
58
59  void CreateEmpty(Stack *S){
60      Top(*S)=-1;
61  }
62
63  bool IsEmpty (Stack S){
64      if(Top(S)==-1){
65          return true;
66      }else{
67          return false;
68      }
69  }
70
```

```
71  bool IsFull(Stack S){
72      if (Top(S)==MaxEl){
73          return true;
74      }else{
75          return false;
76      }
77  }
78
79  void Push(Stack *S,infotype x){
80      if (!IsFull(*S)){
81          if (IsEmpty(*S)){ //jika masih kosong, start dari index ke-1
82              Top(*S)=1;
83          }else{
84              Top(*S)++;
85          }
86          InfoTop(*S)=x;
87      }else{
88          cout<<"Stack Penuh"<<endl;
89      }
90  }
91
92  void Pop(Stack *S,infotype *x){
93      if (!IsEmpty(*S)){
94          *x=InfoTop(*S);
95          if (Top(*S)==1){
96              Top(*S)=-1;
97          }else{
98              Top(*S)--;
99          }
100     }else{
101         cout<<"Stack Kosong"<<endl;
102     }
103 }
104 void Cetak(Stack S){
105     //Cetak Mulai dari Top
106     for (int i=Top(S);i>0;i--){
107         cout<<"["<<S.T[i]<<"]"<<endl;
108     }
109 }
110
```