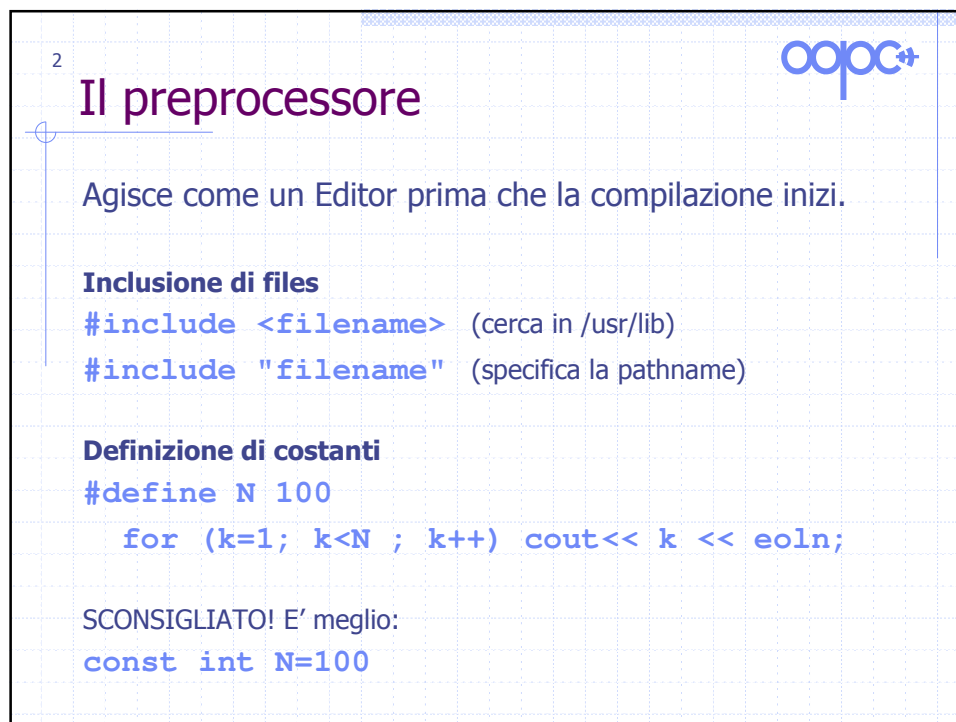
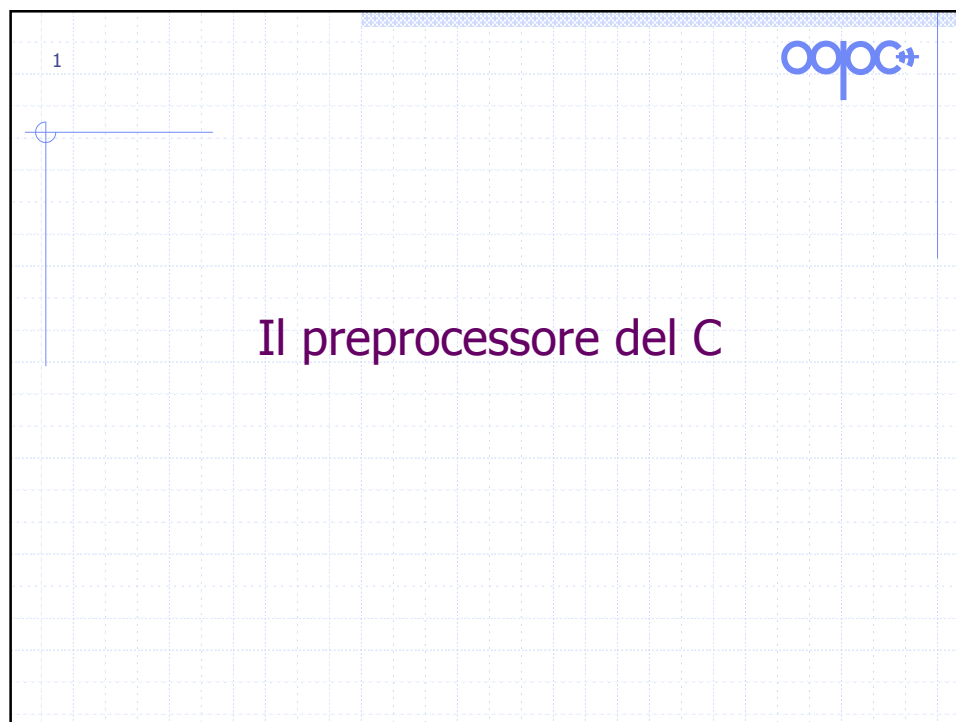




3



Il preprocessore

Compilazione condizionale

```
#define DEBUG  
  
...  
#ifdef DEBUG  
    cout << i <<eoln;  
#endif  
  
...  
#ifndef DEBUG  
    cerr << "VERSIONE UFFICIALE";  
#else  
    cerr << "VERSIONE DI PROVA";  
#endif
```

4



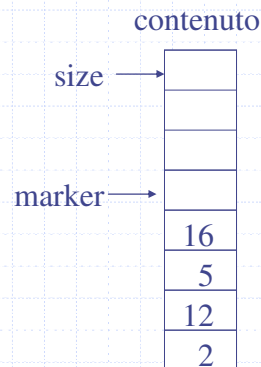
Il nostro esempio guida:
La costruzione di uno stack

5



```
#include <iostream.h>
#include <assert.h>
#define DEBUG
```

```
struct Pila {
    int size;
    int defaultGrowthSize;
    int marker;
    int * contenuto;
};
```



6



```
Pila * crea(int initialSize) {
    //crea una Pila
#ifdef DEBUG
    cout<<"entro in crea"<<endl;
#endif
    Pila * s= new Pila ;
    s->size=initialSize;
    s->defaultGrowthSize=initialSize;
    s->marker=0;
    s-> contenuto=new int[initialSize];
    return s;
}
```

7



```
void distruggi(Pila * s) {  
    //distruggi lo Pila  
    #ifdef DEBUG  
        cout<<"entro in destroy"<<endl;  
    #endif  
    delete [] (s->contenuto);  
    delete s;  
}
```

8



```
void cresci(Pila *s, int increment){  
    //aumenta la dimensione dello Pila  
    #ifdef DEBUG  
        cout<<"entro in cresci"<<endl;  
    #endif  
    s->size+=increment;  
    int * temp=new int[s->size];  
    for (int k=0; k<s->marker;k++) {  
        temp[k]=s->contenuto[k];  
    }  
    delete [] (s->contenuto);  
    s->contenuto=temp;  
}
```

9



```
void inserisci(Pila *s, int k) {  
    //inserisci un nuovo valore  
    #ifdef DEBUG  
        cout<<"entro in inserisci"<<endl;  
    #endif  
    if (s->size==s->marker)  
        cresci(s,s->defaultGrowthSize);  
    s->contenuto[s->marker]=k;  
    s->marker++;  
}
```

10



```
int estrai(Pila *s) {  
    //estrai l'ultimo valore  
    #ifdef DEBUG  
        cout<<"entro in estrai"<<endl;  
    #endif  
    assert(s->marker>0); //precondizione  
    return s->contenuto[--(s->marker)];  
}
```

11



```
void stampaStato(Pila *s) {  
    //stampa lo stato dello Pila  
    cout << "======"<< endl;  
    cout << "size = "<<s->size<<endl;  
    cout << "defaultGrowthSize = "  
        <<s->defaultGrowthSize<<endl;  
    cout << "marker = "<<s->marker <<endl;  
    for (int k=0;k<s->marker;k++)  
        cout << "["<<(s->contenuto[k])<<"]";  
    cout << endl;  
    cout << "======"<< endl;  
}
```

12



```
Pila * copia(Pila * from) {  
    #ifdef DEBUG  
        cout<<"entro in copia"<<endl;  
    #endif  
    Pila * to=crea(from->size);  
    to->defaultGrowthSize=from->defaultGrowthSize;  
    for (int k=0; k<from->marker;k++) {  
        to->contenuto[k]=from->contenuto[k];  
    }  
    to->marker=from->marker;  
    return to;  
}
```

13



```
int main() {
    Pila * s=crea(5);
    cout<<"s"; stampaStato(s);
    for (int k=1; k<10;k++) inserisci(s,k);
    cout<<"s"; stampaStato(s);
    Pila * w = copia(s);
    cout<<"w"; stampaStato(w);
    for (int k=1; k<8;k++)
        cout<<estrai(s)<<endl;
    cout<<"s"; stampaStato(s);
    distruggi(s);
    cout<<"s"; stampaStato(s);
    for (int k=1; k<15;k++)
        cout<<estrai(w)<<endl;
    cout<<"w"; stampaStato(w);
}
```

14



```
bash-2.02$ gcc Pila.cc -lstdc++ -o Pila.exe
```

```
bash-2.02$ Pila.exe
```

```
entro in crea
```

```
s=====
size = 5
defaultGrowthSize = 5
marker = 0
```

```
=====
entro in inserisci
entro in inserisci
entro in inserisci
entro in inserisci
entro in inserisci
entro in inserisci
entro in inserisci
entro in inserisci
entro in inserisci
```

```
s=====
size = 10
defaultGrowthSize = 5
marker = 9
[1][2][3][4][5][6][7][8][9]
```

```
entro in copia
```

```
w=====
size = 10
defaultGrowthSize = 10
marker = 9
[1][2][3][4][5][6][7][8][9]
```

15

entro in estrai
9
entro in estrai
8
...
entro in estrai
4
entro in estrai
3
s=====
size = 10
defaultGrowthSize = 5
marker = 2
[1][2]
=====

entro in distruggi
s=====
size = 1627775824
defaultGrowthSize = 1627775824
marker = 2
[1627775848][1627775848]
=====
entro in estrai
9
entro in estrai
8
...
entro in estrai
2
entro in estrai
1
entro in estrai
assertion "s->marker>0" failed: file "Pila.cc", line 72
bash-2.02\$

