

Moduli

Esempio: la pila

Modulo Pila

- Il **modulo** Pila è costituito da due file, uno^(*) di definizione (.cpp) ed uno di intestazione (.h)
 - File di definizione: [pila.cpp](#)
 - File di intestazione: [pila.h](#)
 - I file [pila.cpp](#) e [pila.h](#) costituiscono un'unità di compilazione
- Il modulo Pila offre i seguenti **servizi** (è un modulo **servitore**)
 - Il tipo [struct Pila](#)
 - La costante [N](#)
 - Le operazioni: [inizializza](#), [push](#), [pop](#), [top](#), [vuoto](#), [pieno](#)

(*) Il generale, uno o piú

Il file di definizione pila.cpp

```
#include "pila.h"
```

```
void inicializza(Pila& s) { s.tt = 0; }
```

```
void push(Pila& s, int v) { s.ee[s.tt++] = v; }
```

```
int pop(Pila& s) { return s.ee[--s.tt]; }
```

```
int top(Pila s) { return s.ee[s.tt - 1]; }
```

```
int vuoto(Pila s) { return s.tt <= 0; }
```

```
int pieno(Pila s) { return s.tt >= N; }
```

Il file di intestazione pila.h

```
const int N = 10;    // capacità della pila
struct Pila {
    int tt;           // indice del primo elemento vuoto
    int ee[N];       // elementi della pila
};
void inizializza(Pila&);
void push(Pila&, int);
int pop(Pila&);
int top(Pila);
int vuoto(Pila);
int pieno(Pila);
```

Modulo cliente

- Il modulo cliente main utilizza i servizi offerti dal modulo pila
- Il modulo cliente main è costituito da main.cpp e pila.h che costituiscono un'unità di compilazione

File main.cpp

```
#include <iostream>
#include "pila.h"
using namespace std;
int main()
{
    Pila p;
    inizializza(p);
    push(p, 1); push(p, 2);
    cout << pop(p) << endl;
}
```