```
generic
  max_elem: natural;
  type elem is private;
package d_pila is

  type pila is limited private;

  procedure buida(p: in out pila);
  procedure empila(p: in out pila; e: in elem);
  procedure desempila(p: in out pila);
  function  cim(p: in pila) return elem;
  function  es_buida(p: in pila) return boolean;

  limit_de_capacitat, mal_us: exception;

private

  min_elem: constant natural:= 0;

  subtype rang_pila is Natural range min_elem..max_elem;
  type memoria is array (rang_pila) of elem;

  type pila is
    record
      idx: rang_pila;
      mem: memoria;
    end record;

end d_pila;



--BODY
package body d_pila is
  procedure buida(p: in out pila) is
  begin
    p.idx:= min_elem;
  end buida;


  procedure empila(p: in out pila; e: in elem) is
  begin
    if p.idx = max_elem then raise limit_de_capacitat; end if;
    p.idx:= p.idx + 1;
    p.mem(p.idx):= e;
  end empila;


  procedure desempila(p: in out pila) is
  begin
    if p.idx = min_elem then raise mal_us; end if;
    p.idx:= p.idx - 1;
  end desempila;


  function cim(p: in pila) return elem is
  begin
    if p.idx = min_elem then raise mal_us; end if;
    return p.mem(p.idx);
  end cim;


  function es_buida(p: in pila) return boolean is
  begin
    return p.idx = min_elem;
  end es_buida;

end d_pila;
```