

monitor-stack

Si implementi, tramite monitor, uno stack a capacità limitata N con chiamate:

- void push(Object o) - inserisce un elemento nello stack (attende se pieno)
- Object pop() - rimuove e restituisce un elemento dallo stack (attende se vuoto)
- Object top() - restituisce il prossimo elemento da rimuovere nello stack, senza rimuoverlo (attende se vuoto)
- boolean empty() - restituisce true se stack è vuoto, falso altrimenti
- void atleast(int n) - attende che nello stack ci siano almeno n elementi

```

monitor Stack {

    Object[] buffer;
    int size;

    condition notfull, notempty;
    condition topready;
    condition[] notsize;

    Stack(int N)
    {
        buffer = new Object[N];
        notsize = new condition[N];
        size = 0;
    }

    p.e. void push(Object o)
    {
        if (size == N) {
            notfull.wait();
        }
        buffer[size++] = o;
        notsize[size].signal();
        topready.signal();
        notempty.signal();
    }

    p.e. Object pop()
    {
        if (size == 0) {
            notempty.wait();
        }
        Object ret = buffer[--size];
        notfull.signal();
        return ret;
    }

    p.e. Object top()
    {
        if (size == 0) {
            topready.wait();
        }
        topready.signal();
        return buffer[size-1];
    }

    p.e. void atLeast(int n) {
        if (size < n-1) {
            notsize[n-1].wait();
        }
        notsize[n-1].signal();
    }
}

```