

```

NECESARIO "queue_eda.h"

#include <iostream>
#include <fstream>
#include "queue_eda.h"
using namespace std;

template<class T>
class queue_plus : public queue<T> {
    using Nodo = typename queue<T>::Nodo;

public:

    int saltar(const int N) {
        Nodo* nodo = this->prim;
        Nodo* anterior = this->prim;
        int n;
        int i = this->nelems;
        this->ult->sig = this->prim;

        while (i > 1) {
            for (int j = 0; j < N; j++) {
                anterior = nodo;
                nodo = nodo->sig;
            }
            anterior->sig = nodo->sig;
            nodo = nodo->sig;
            i--;
        }

        n = nodo->elem;

        this->nelems = 0;
        this->prim = this->ult = nullptr;
        return n;
    }
};

bool resuelveCaso() {
    queue_plus<int> alumnos;
    int n_alumnos, n_salta;
    int alumno = 0;

    cin >> n_alumnos >> n_salta;

    if (n_alumnos == 0 && n_salta == 0)
        return false;

    for (int i = 0; i < n_alumnos; i++) {

```

```
        alumnos.push(i + 1);
    }

    alumno = alumnos.saltar(n_salta);

    cout << alumno << endl;

    return true;
}

int main() {
    // ajustes para que cin extraiga directamente de un fichero
#ifndef DOMJUDGE
    std::ifstream in("datos.txt");
    auto cinbuf = std::cin.rdbuf(in.rdbuf());
#endif

    while (resuelveCaso());

    // para dejar todo como estaba al principio
#ifndef DOMJUDGE
    std::cin.rdbuf(cinbuf);
    system("PAUSE");
#endif
    return 0;
}
```