

Zadanie 10

Proszę napisać (i przetestować) szablon klasy `Queue` reprezentującej kolejkę (FIFO) elementów pewnego typu. Implementacja powinna opierać się na strukturze listy jednokierunkowej, dla której pamiętany jest wskaźnik `head` wskazujący na pierwszy element listy oraz `tail` wskazujący na element ostatni. Operacje, jakie są określone dla kolejki są następujące:

- konstruktor tworzący kolejkę pustą;
- `put` — dodaje element na końcu kolejki (bez przebiegania przez wszystkie elementy listy!);
- `get` — zwraca element z pierwszego węzła kolejki i usuwa ten węzeł;
- `empty` — odpowiada na pytanie, czy kolejka jest pusta;
- destruktor (który powinien coś pisać, abyśmy widzieli, które elementy są usuwane)..

Program może mieć następującą strukturę:

```
#include <iostream>

template<typename T>
class Queue {
    struct Node {
        // ...
    };
    Node* head;
    Node* tail;
public:
    Queue();
    bool empty() const;
    void put(const T& data);
    T get();
    ~Queue();
};

int main() {
    int data1, data2;

    Queue<int>* q = new Queue<int>();
    q->put(1);
    data1 = q->get();
    std::cout << " data1=" << data1 << std::endl;
```

```

q->put(1);
q->put(2);
data1 = q->get();
data2 = q->get();
std::cout << " data1=" << data1
          << " data2=" << data2 << std::endl;

q->put(1); q->put(2); q->put(3);
q->put(4); q->put(5); q->put(6);
while (!q->empty()) std::cout << " " << q->get();
std::cout << std::endl;

q->put(1); q->put(2); q->put(3);
delete q;
}

```

Program powinien wydrukować coś w rodzaju:

```

data1=1
data1=1 data2=2
1 2 3 4 5 6
Del:1 Del:2 Del:3

```

Termin: do 1 lutego (włącznie)

Rozwiązania, w postaci **jednego** pliku źródłowego zawierającego treść programu, proszę wrzucać w systemie EDU do katalogu „Foldery zadań / Zadanie_XX”, gdzie 'XX' jest numerem zadania.

Nazwą pliku powinno być nazwisko z dużej litery (bez polskich znaków); rozszerzeniem musi być '.cpp', czyli np. *Malinowska.cpp*.