

Przyjąć, że udostępniona jest przestrzeń nazw std, std::placeholders i boost**Zadanie 1 (8pkt)**

Tekst jest dostępny do odczytu po podaniu identyfikatora, który jest liczbą dodatnią. Dla wybranych tekstów funkcja `calculate` dostarcza kategorie tekstów, tak jak pokazano poniżej. Zmień kod, aby użyć dwóch wątków obliczeniowych, w jednym powinna się odbywać ekstrakcja słów kluczowych z tekstu (funkcja `calc_keywords`), a w drugim obliczanie kategorii na podstawie słów kluczowych (funkcja `calc_category`).

```
using Keywords = vector<string>; Keywords calc_keywords(int id); string calc_category(const Keywords& k);
```

```
class ThreadCalc {
public:
    ThreadCalc(const vector<int>& ids) :ids_(ids) {}
    void operator() () {
        for(int id : ids_) {
            Keywords k = calc_keywords(id); //expensive
            string c = calc_category(k); //expensive
            out_[id] = c;
        }
    }
    const map<int,string>& getResult() { return out_; }
private:
    vector<int> ids_;
    map<int,string> out_;
};
map<int,string> calculate(const vector<int>& ids) {
    ThreadCalc calc(ids);
    thread thrd(ref(calc));
    thrd.join();
    return calc.getResult();
}
```

Zadanie 2 (5pkt)

- 1) Zaimplementować szablon `max`, który dostarczy wartość maksymalną z tablicy `std::array<T,N>`, gdzie T jest typem liczbowym, zaś N jest rozmiarem.
- 2) Postaraj się, aby szablon nie używał pętli do obliczeń. N będzie małe.
- 3) Zadbaj, aby można było wołać szablon tylko dla typów numerycznych.

`template <class T> struct is_arithmetic` bada, czy typ jest typem numerycznym, jeżeli tak, to `is_arithmetic<T>::value` jest typu `bool` i ma wartość `true`, w przeciwnym wypadku `is_arithmetic<T>::value` ma wartość `false`.

Notatki / uwagi do prowadzącego

