
Przyjąć, że udostępniona jest przestrzeń nazw std i boost

W rozwiązaniach można używać dowolnych udogodnień z biblioteki standardowej C++ oraz bibliotek Boost.

Zadanie 1 (2pkt)

Poniżej używamy obiektów `Calc`. Użycia nie można poprawić. Popraw kod klasy `Calc` (jest w ramce), jeżeli jest możliwość wystąpienia wyścigu lub blokady.

```
int main() {
    Calc c1, c2, c3;
    c1.setIn(&c3);
    c2.setIn(&c1);
    c3.setIn(&c2);
    thread thrd1( ref(c1) ), thrd2( ref(c2) ), thrd3( ref(c3) );
    thrd1.join(); thrd2.join(); thrd3.join();
}
```

```

class Calc {
public:
    static const int TOKEN_END = 100;
    static const int NUM = 100;
    Calc() : in_(nullptr) {
        data_.push_back(0); //data is not empty at start
        data_.push_back(0);
    }
    void setIn(Calc* p) { in_ = p; }
    int read() {
        int ret = TOKEN_END;
        lock_guard lock(m_);
        if(!data_.empty() ) {
            ret = data_.front();
            data_.pop_front();
        }
        return ret;
    }
    void operator()() {
        for(int i=0;i<NUM;++i) {
            //tutaj sa obliczenia (kod nieistotny dla zadania)
            lock_guard lock(m_);
            if(in_) {
                int x = in_>read();
                if( x != TOKEN_END) {
                    //tutaj sa obliczenia (kod nieistotny dla zadania)
                    data_.push_back(x + 1);
                }
            }
        }
    }
};

private:
    Calc* in_;
    list<int> data_;
    mutex m_;
};

```

Pytanie 1 (1pkt)

Jakie treści proponujesz dodać do ZPR w przyszłych edycjach?

Ile godzin poświęciłeś na ZPR (uwzględnij wszystko, tzn. wykład, kolokwia, projekt i inne)?

?	
---	--

Zadanie 2 (2pkt)

Napisz funkcję `second_best`, zwracającą nazwisko osoby, która pojawiłaby się na drugim miejscu, jeżeli posortowalibyśmy malejąco zawodników (obiekty typu `Competitor`) po punktach (najwięcej punktów na pierwszym miejscu). Jeżeli wynik nie jest jednoznaczny (np. wszyscy mają tyle samo punktów) to zwraca nazwisko dowolnej osoby spełniającej warunki. Nie ma miejsca na tworzenie kopii kolekcji.

Dla przykładu poniżej powinno wydrukować `Dabacki`.

```
struct Competitor {
    string name;
    int points;
};
using Competitors = vector<Competitor>;

Competitors c;
c.push_back( { "Abacki", 100 } );
c.push_back( { "Babacki", 80 } );
c.push_back( { "Cabacki", 90 } );
c.push_back( { "Dabacki", 99 } );
cout << second_best(c) << endl;
```

```
std::string second_best(const Competitors& competitors) {
```

Zadanie 3 (1pkt)

Podaj przykład użycia klas cech (traits) w C++. Może być kod, może być opis.

Uwagi do prowadzącego (R.Nowak):

Zadanie 4 (2pkt)

Zaproponuj reprezentację typu `Command` oraz napisz funkcję `TurtleState executeProgram(Program)`. Funkcja ta interpretuje program języka LOGO i aktualizuje stan „żółwia”, który reprezentowany jest przez strukturę:

```
using Command = /* ??? */;
using Program = std::vector<Command>;
struct TurtleState {
    std::pair<double, double> pos;
    double angle{0};
    Color color{Color::Black};
    bool penDown{true};
};
```

Można założyć istnienie funkcji `newPosFrom(std::pair<double, double>, double angle, double distance)` służącej do obliczania nowej pozycji żółwia. Polecenia reprezentowane są przez struktury:

```
struct Move { double distance; }; struct Turn { double angle; };
struct PenUp {};
struct PenDown {};
struct SetColor { enum class Color color; };
```

Zadanie 5 (2pkt)

Przy pomocy algorytmów biblioteki standardowej zaimplementuj funkcję (nie szablon funkcji!), którą można wywołać jak poniżej:

```
std::vector<int> v{...};  
std::array<int, 10> a{...};  
int result = calc(v) + calc(a);
```

Funkcja ma działać, jak następuje:

- jeżeli wszystkie liczby są dodatnie, to zwraca liczbę elementów, które są nieparzyste;
- w przeciwnym razie, jeżeli argument funkcji posiada przynajmniej 4 elementy, to funkcja zwraca czwarty najmniejszy element;
- w przeciwnym razie zwraca zero.

Funkcja nie może użyć pętli for, while ani std::for_each. Funkcja może zmodyfikować swój argument.

Uwagi do prowadzącego (W. Wysota):