

Zadanie 1

Napisz klasę **Letter**, która będzie służyć do analizowania tekstów. Obiekty klasy zawierają tablicę 26 liczb typu **int** odpowiadających literom alfabetu angielskiego (element o indeksie 0 — literze 'A', ... , o indeksie 25 — literze 'Z'). Konstruktor klasy pobiera tekst (jako wskaźnik typu **const char***) i wypełnia tablicę liczbami wystąpień poszczególnych liter.

Uwaga: litery duże i odpowiadające im małe są uważane za takie same.

Klasa udostępnia metody

- **getMostFrequent** zwracającą znak, który występuje najczęściej;
- **numOfDifferent** zwracającą liczbę różnych liter;
- **printFrequencies** drukującą „histogram” opisujący liczby wystąpień poszczególnych liter: wyrównany od dołu, z kolumnami złożonymi z tyłu gwiazdek, ile wynosiła liczba wystąpień odpowiedniej litery (patrz przykład poniżej).

Na przykład następujący program

```
#include <iostream>

class Letters {
    int letters[26]{0};
    // ...
};

int main() {
    const char* text =
        "To be, or not to be- that is the questiona\n:"
        "Whether 'tis nobler in the mind to suffer\n"
        "The slings and arrows of outrageous fortune\n"
        "Or to take arms against a sea of troubles,\n"
        "And by opposing end them.";
    Letters lett(text);
    std::cout << "Most frequent letter: "
               << lett.getMostFrequent() << std::endl;
    std::cout << "Number of different letters: "
               << lett.numOfDifferent() << std::endl;
    std::cout << "Frequency table of all letters:\n";
    std::cout << std::endl;
    lett.printFrequencies();
}
```

powinien wydrukować

[illegible]

Napisz i przetestuj funkcję pobierającą

- a zwracająca wskaźnik do funkcji, również typu **double** \rightarrow **double**.

Aby znaleźć maksimum funkcji na odcinku, można „przejechać” przez ten odcinek z małym krokiem (np. $\epsilon = 10^{-4}$) i znajdować w każdym punkcie wartość funkcji.

W programie testowym można użyć wskaźników do funkcji bibliotecznych z nagłówka ***cmath*** (jak sin, cos, exp) oraz przynajmniej jednej funkcji napisanej samemu.

```
#include <iostream>
#include <cmath>
```

```

using namespace std;

double f1(double x) { return x/4; }
double f2(double x) { return -2*x; }

using D2D = double (*)(double);

D2D maxfun(D2D funcs[], int size,
           double a, double b, double* pmax) {
    static const double eps = 1e-4;
    // ...
}

int main() {
    double xmax;
    D2D funcs[] = {f1, sin, f2};
    D2D pf = maxfun(funcs, 3, 0, 3, &xmax);
    cout << "xmax = " << xmax << "; f(xmax) = "
         << pf(xmax) << endl;
}

```

powinien wydrukować

```
xmax = 1.5708; f(xmax) = 1
```

(bo z trzech funkcji **f1**, **sin** i **f2** na odcinku $[0, 3]$ największą wartość, równą 1, przyjmuje druga z nich dla argumentu $\pi/2$).
