

Zadanie 6

Zdefiniować pięć funkcji operujących na C-napisach (i jedną, **isLetter**, na pojedynczym znaku):

```
size_t length(const char* cstr);           // 1
bool  isLetter(char c);                    // 2
char*  reverse(char* cstr);                // 3
size_t words(const char* cstr);            // 4
size_t words2(const char* cstr);           // 5
char*  concat(char* t, const char* s);     // 6
```

gdzie

1. **length** zwraca długość napisu (nie licząc znaku `'\0'` na jego końcu);
2. **isLetter** sprawdza, czy przekazany znak `c` jest dużą lub małą literą (nie używać znajomości kodów ASCII);
3. **reverse** odwraca kolejność znaków w przekazanym napisie i zwraca niezmienioną wartość przekazanego wskaźnika `cstr`;
4. **words** zwraca ilość słów w przekazanym napisie, przy czym za słowo uważamy niepusty ciąg *liter*, dużych i małych, taki, że bezpośrednio przed nim i za nim nie ma litery;
5. **words2** zwraca ilość słów w przekazanym napisie, ale tym razem liczymy tylko „słowa” co najmniej dwuliterowe;
6. **concat** „dokleja” do napisu `t` (od target) napis `s` (od source); oczywiście trzeba zadbać o to, aby zarezerwowany pod adresem `t` obszar pamięci wystarczał na pomieszczenie obu napisów wraz z kończącym znakiem `'\0'`! Funkcja zwraca niezmodyfikowaną wartość pierwszego argumentu.

UWAGA: wszystkie te funkcje powinny być zaimplementowane samodzielnie, bez odwoływania się do funkcji z biblioteki standardowej — w szczególności **strlen**, **isupper**, **isalpha**, **strcpy** itd. Nie włączaj żadnych plików nagłówkowych z wyjątkiem **iostream**. Nie twórz żadnych pomocniczych tablic.

Na przykład program:

```
#include <iostream>
```

[download CStrings.cpp](#)

```
size_t length(const char* cstr);
bool  isLetter(char c);
char*  reverse(char* cstr);
size_t words(const char* cstr);
size_t words2(const char* cstr);
char*  concat(char* t, const char* s);
```

```

int main() {
    using std::cout; using std::endl;
    char s1[] = "Alice in Wonderland";
    cout << "reverse: " << reverse(s1) << endl;
    cout << "length : " << length(s1) << endl;

    char s2[] = " ... for (int i = 0; i < n; ++i){...} ...";
    cout << "words : " << words(s2) << endl;
    cout << "words2 : " << words2(s2) << endl;

    char s3[100] = "Hello";
    cout << "concat : "
        << concat(concat(s3, " world"), "!") << endl;
}

// definitions of functions

```

powinien wydrukować

```

reverse: dnaIrednoW ni ecilA
length : 19
words   : 6
words2  : 2
concat  : Hello, world!

```

Termin: do 13 maja (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*.