

Saskaitīt vārdus

Hoola-Hoola girls like hooligans → 5

```
#include <iostream>
#include <string>

using namespace std;

int countWords(const string& str)
{
    int cnt = 0;
    for (int i = 0; str[i]; i++)
        if (isalpha(str[i]) && !isalpha(str[i + 1])) cnt++;
    return cnt;
}

int main()
{
    string str;
    cout << "Enter a string: "; getline(cin, str);
    cout << "Amount of words: " << countWords(str) << endl;
}
```

legūt vārdu sarakstu

Hoola-Hoola girls like hooligans → hoola, girls, like, hooligans

```
string toLower(const string& str)
{
    string res {" "};
    for (char chr : str) res += tolower(chr);
    return res;
}

void appendWord(const string& word, string words[], int &wordCount)
{
    for (int i = 0; i < wordCount; i++) if (words[i] == word) return;
    words[wordCount++] = word;
}

void showWords(string words[], int wordCount)
{
    for (int i = 0; i < wordCount; i++) {
        cout << words[i];
        if (i < wordCount - 1) cout << ", ";
    }
}
```

legūt vārdu sarakstu

Hoola-Hoola girls like hooligans → hoola, girls, like, hooligans

```
void getWords(const string& str, string words[], int &wordCount)
{
    wordCount = 0;
    string word {" "};
    for (char chr : str) {
        if (isalpha(chr)) word += chr;
        else if (word != " ") {
            appendWord(toLower(word), words, wordCount);
            word = " ";
        }
    }
    if (word != " ") appendWord(toLower(word), words, wordCount);
}
```

legūt vārdu sarakstu

Hoola-Hoola girls like hooligans → hoola, girls, like, hooligans

```
#include <iostream>
#include <string>

using namespace std;

string toLower(const string str) {...}
void appendWord(const string word, string words[], int &wordCount) {...}
void showWords(string words[], int wordCount) {...}
void getWords(const string str, string words[], int &wordCount) {...}

int main()
{
    string str;
    cout << "Enter a string: "; getline(cin, str);
    string words[100]; int count = 0;
    getWords(str, words, count);
    cout << "Words: ";
    showWords(words, count);
    cout << endl;
}
```

Mājas uzdevums: vārdu rinda

Pieprasīt no lietotāja simbolu rindu. Pieņemot par vārdu jebkuru latīņu burtu secību, risināt sekojošus uzdevumus:

- A:** saskaitīt vārdus, kas satur tikai mazos burtus;
- B:** noteikt visgarākā vārda garumu;
- C:** mainiet katra vārda pirmo burtu uz lielo, bet pārējos uz mazajiem;
- D:** iegūt vārdu sarakstu, kas ir iekļauti apaļās iekavās (ņemot vērā, ka iekavas var būt ligzdotas).

Piemērs 1:

`Strings can contain single-byte (ASCII) or multi-byte (UNICODE) characters`

- A:** 8
- B:** 10
- C:** Strings Can Contain Single-Byte (Ascii) Or Multi-Byte (Unicode) Characters
- D:** ascii, unicode

Piemērs 2:

`Paradigm (PP(procedural program), OOP(object oriented program)) is program style`

- A:** 8
- B:** 10
- C:** Paradigm (Pp(Procedural Program), Oop(Object Oriented Program)) Is Program Style
- D:** pp, procedural, program, oop, object, oriented