Object Oriented programming and software engineering – Lab 2

Adam Korytowski - 2025

1. File support (#include <fstream>)

- Before you start working on the file it is necessary to create a variable that will allow us to perform operations on the selected file:

```
std::fstream file;
```

- The created variable does not point to any file. To assign a specific file to it, use the open() function from class std::fstream:

```
file.open( "file_name.txt", file_open_mode);
```

When assigning a file to a variable, specify one or more file opening modes:
ios::app — Sets the internal pointer to the file at its end. File open in write-only mode. Data can only be saved at the end of the file.
ios::ate — Sets the internal pointer to the end of the file at the time the file opens.
ios::binary — Information for the compiler for data to be treated as a binary data stream.
ios::in — Allows reading data from a file.
ios::out — Allows writing data to a file.
ios::trunc — File content is cleared when opening.
ios::in | ios::out – Clears the file before opening

```
fstream file;
file.open( "name_of_file.txt", ios::in | ios::out );
```

When the operation with the file ends, close the file. For this purpose, the function close() will be used:

```
file.close();
```

Example of opening a text file:

```
vector<string>fileContent;
fstream file;
file.open("products.txt", ios::in | ios::out);
if (file.good() == true)
{
    if (file.is_open())
    {
        //std::cout << file.rdbuf(); //prints content of the file
        std::string word;

        while (file >> word)
        {
            fileContent.push_back(word);
        }
    }
}
```

Example of saving a text file:

```
std::ofstream outputFile("testOutput.txt");
for (const auto& word : fileContent)
{
    outputFile << word << endl;
}
outputFile.close();
```

2. Write a program with requirements (2 pts each)

Program requirements:

- The program loads a list of “products” along with the number of available items, from an external text or binary file at startup and each time the list is refreshed.
- Once purchase have been made, the number of products is refreshed and saved to an external file.
- If any of the products is exhausted, a note about the need to replenish the product is added to the external file
- At the start of the program (and only at start-up), if an annotation about the need to replenish a product is found, its quantity is increased to 10 units.