

Politechnika Śląska
Wydział Informatyki, Elektroniki i Informatyki

Programowanie Komputerów
Informatyka, rok 1 Katowice

„Wykaz sprzętu w firmie z podziałem na działy.”

Autor

Maciej Fajlhauer
Gr. 1/1

1 Wstęp	1
1.1 Treść zadania	1
1.2 Plik testowy	1
1.3 Instrukcja	1
1.4 Schemat blokowy projektu	2
1.5 Diagram klas	2
1.6 <u>Zagadnienia z laboratorium w projekcie</u>	2
 2 Indeks klas	 3
2.1 Lista klas	3
 3 Indeks plików	 5
3.1 Lista plików	5
 4 Dokumentacja klas	 7
4.1 Dokumentacja klasy Equipment	7
4.1.1 Opis szczegółowy	7
4.1.2 Dokumentacja konstruktora i destruktora	7
4.1.2.1 Equipment()	8
4.1.3 Dokumentacja funkcji składowych	9
4.1.3.1 getDepartment()	9
4.1.3.2 getId()	9
4.1.3.3 getName()	9
4.1.3.4 setDepartment()	9
4.1.3.5 setName()	10
4.2 Dokumentacja klasy EquipmentDatabase	10
4.2.1 Opis szczegółowy	11
4.2.2 Dokumentacja konstruktora i destruktora	11
4.2.2.1 EquipmentDatabase()	11
4.2.3 Dokumentacja funkcji składowych	11
4.2.3.1 addEquipment()	11
4.2.3.2 getEquipmentByDepartment()	11
4.2.3.3 getEquipmentList()	12
4.2.3.4 modifyEquipment()	12
4.2.3.5 removeEquipment()	12
4.2.3.6 searchEquipment()	12
 5 Dokumentacja plików	 15
5.1 Dokumentacja pliku dbfunction.cpp	15
5.1.1 Dokumentacja funkcji	15
5.1.1.1 addNewEquipment()	16
5.1.1.2 dbcopy()	16
5.1.1.3 displayEquipmentByDepartment()	16
5.1.1.4 displayEquipmentList()	16

5.1.1.5 loginAsAdmin().....	16
5.1.1.6 logowanie().....	17
5.1.1.7 modifyEquipment()	17
5.1.1.8 option().....	17
5.1.1.9 removeEquipment().....	17
5.1.1.10 searchEquipment()	17
5.2 Dokumentacja pliku header.h	17
5.2.1 Dokumentacja funkcji	18
5.2.1.1 addNewEquipment().....	18
5.2.1.2 displayEquipmentByDepartment()	18
5.2.1.3 displayEquipmentList().....	19
5.2.1.4 loginAsAdmin().....	19
5.2.1.5 logowanie().....	19
5.2.1.6 modifyEquipment()	19
5.2.1.7 option().....	19
5.2.1.8 removeEquipment().....	20
5.2.1.9 searchEquipment()	20
5.3 header.h	20
5.4 Dokumentacja pliku main.cpp.....	21
5.4.1 Dokumentacja funkcji	21
5.4.1.1 main().....	21
Skorowidz	23

Wstęp

1.1 Treść zadania

Napisac program realizujący zadania związane z zarządzaniem i wyświetlaniem spisu sprzętu w firmie.

1.2 Plik testowy

W załączonych plikach znajduje się plik tekstowy „db.txt” zawierający 60 wierszy. Plik ten jest reprezentacją bazy danych sprzętu w firmie. Każdy wiersz ma następującą budowę: <unikatowy numer id> <nazwa_sprzetu> <nazwa_←, dzialu>

1.3 Instrukcja

Po uruchomieniu programu oraz podaniu nazwy pliku zawierającego bazy danych, użytkownik ma do wyboru dwie opcje:

- 1) Wyświetlenie pełnego wykazu sprzętu - ta opcja pozwala na wyświetlenie pełnej listy sprzętu znajdującego się w firmie.
- 2) Zalogowanie się na konto Admin w celu zdobycia większej ilości uprawnień opcja umożliwia użytkownikowi dostęp do konta administracyjnego i dodatkowych funkcjonalności.

Hasło pozwalające na przełączenie konta z domyślnego na administracyjne to **"Pk2ktW"**.

Uwaga: Wielkość i kolejność liter mają znaczenie!

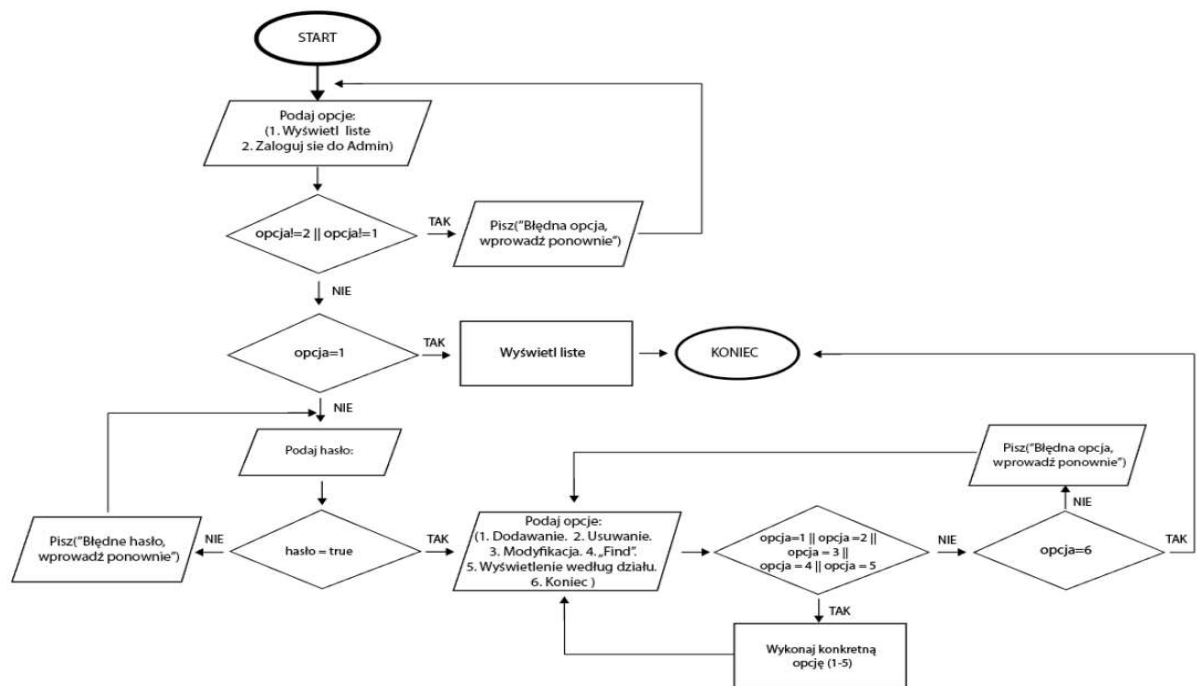
Po zalogowaniu się na konto Admin, użytkownik może korzystać z następujących opcji:

- a) Dodawanie - ta opcja pozwala na dodawanie nowego sprzętu do bazy danych.
- b) Usuwanie - ta opcja pozwala na usuwanie konkretnego sprzętu/ przedmiotu z bazy danych.
- c) Modyfikacja - ta opcja umożliwia modyfikowanie informacji o konkretnym sprzęcie znajdującym się w bazie danych.
- d) Opcja find - ta opcja pozwala na wyszukiwanie sprzętu w bazie danych za pomocą nazwy lub numeru ID.
- e) Wyświetlenie względem konkretnego działu w firmie - ta opcja umożliwia wyświetlenie sprzętu związanego z danym działem w firmie.

Autor

Maciej Fajlhauer

1.4 Schemat blokowy



1.5 Diagram klas

Equipment
<ul style="list-style-type: none"> - id: int - name: string - department: string
getId(): int getName(): string getDepartment(): string setName(name: string): void setDepartment(department: string): void



EquipmentDatabase
equipment: vector
getEquipmentList(): vector<Equipment> addEquipment(equipment: Equipment): void removeEquipment(id: int): void modifyEquipment(id: int, name: string, department: string): void searchEquipment(query: string): vector<Equipment> getEquipmentByDepartment(department: string): vector<Equipment>

1.6 Zagadnienia z labolatorium w projekcie

1. `Equipment` i `EquipmentDatabase` są zdefiniowane jako **klasy**, które przechowują informacje o sprzęcie i bazie danych sprzętu.

2. Klasa `Equipment` posiada **konstruktor** `Equipment(int id, string name, string department)`, który inicjalizuje obiekty klasy z przekazanymi parametrami. Klasa `EquipmentDatabase` posiada **konstruktor** domyślny `EquipmentDatabase()`, który tworzy obiekt bazy danych.

3. **Metody dostępne:** Klasy `Equipment` i `EquipmentDatabase` posiadają metody dostępne (getter) do odczytu wartości pól obiektów

4. **Metody modyfikujące:** Klasy `Equipment` i `EquipmentDatabase` posiadają metody modyfikujące (setter), które umożliwiają zmianę wartości pól obiektów.

5. Klasa `EquipmentDatabase` przechowuje **wektor** obiektów klasy `Equipment` w celu przechowywania danych sprzętu.

6. W różnych metodach klasy `EquipmentDatabase` wykorzystywane są **pętle do iteracji po elementach wektora** sprzętu i wykonywania operacji na tych elementach.

7. Metoda `removeEquipment(int id)` klasy `EquipmentDatabase` wykorzystuje **lambdę** jako argument funkcji `remove_if` do filtrowania elementów wektora sprzętu.

8. **Inne zagadnienia:** W kodzie występują również funkcje pomocnicze, takie jak `displayEquipmentList`, `loginAsAdmin`, `dbcopy`, `logowanie` i `option`, które zarządzają **interakcją z użytkownikiem i operacjami na bazie danych sprzętu**.

Dodatkowo zadbałem o przekazywanie argumentów funkcji przez referencję oraz użyłem w programie takich elementów jak: `system("pause");` oraz `system("cls");` w celu stworzenia bardziej przejrzystego środowiska pracy.

1.7 Co warto poprawić w przyszłości, czego nie zrobiłem.

Elementem wartym dopracowania jest próba implementacji większej ilości materiału poznanego na labolatoriach. Odnośnie samego projektu, dobrym pomysłem, którego nie udało mi się zrealizować była by funkcja tworząca kopie zapasową bazy po zakończonej pracy.

Rozdział 2

Indeks klas

2.1 Lista klas

Equipment	7
EquipmentDatabase	10

Rozdział 3

Indeks plików

3.1 Lista plików

Tutaj znajduje się lista wszystkich plików z ich krótkimi opisami:

dbfunction.cpp	15
header.h	17
main.cpp	21

Rozdział 4

Dokumentacja klas

4.1 Dokumentacja klasy Equipment

Klasa Equipment reprezentuje sprzęt. Ma prywatne pola `id_`, `name_` i `department_`, które są dostępne za pomocą publicznych metod `getId()`, `getName()` i `getDepartment()`. Klasa Equipment ma również publiczne metody `setName()` i `setDepartment()`, które pozwalają na modyfikowanie nazwy i działu sprzętu.

```
#include <header.h>
```

Metody publiczne

- Equipment (int id, string name, string department)
- int getId () const
Zwraca identyfikator sprzętu.
- string getName () const
Zwraca nazwę sprzętu.
- string getDepartment () const
Zwraca nazwę działu, do którego przynależy sprzęt.
- void setName (string name)
Metoda publiczna umożliwiająca ustawienie nazwy sprzętu.
- void setDepartment (string department)
Metoda publiczna umożliwiająca ustawienie nazwy działu, do którego przynależy sprzęt.

4.1.1 Opis szczegółowy

Klasa Equipment reprezentuje sprzęt. Ma prywatne pola `id_`, `name_` i `department_`, które są dostępne za pomocą publicznych metod `getId()`, `getName()` i `getDepartment()`. Klasa Equipment ma również publiczne metody `setName()` i `setDepartment()`, które pozwalają na modyfikowanie nazwy i działu sprzętu.

4.1.2 Dokumentacja konstruktora i destruktor

4.1.2.1 Equipment()

```
Equipment::Equipment (
    int id,
    string name,
    string department )
```

Konstruktor klasy Equipment

Parametry

<i>id</i>	unikatowy numer id sprzętu
<i>name</i>	nazwa sprzętu
<i>department</i>	nazwa działu do którego przynależy sprzęt

4.1.3 Dokumentacja funkcji składowych**4.1.3.1 getDepartment()**

```
string Equipment::getDepartment ( ) const
```

Zwraca nazwę działu, do którego przynależy sprzęt.

Zwraca

Nazwa działu sprzętu.

4.1.3.2 getId()

```
int Equipment::getId ( ) const
```

Zwraca identyfikator sprzętu.

Zwraca

Identyfikator sprzętu.

4.1.3.3 getName()

```
string Equipment::getName ( ) const
```

Zwraca nazwę sprzętu.

Zwraca

Nazwa sprzętu.

4.1.3.4 setDepartment()

```
void Equipment::setDepartment (
    string department )
```

Metoda publiczna umożliwiająca ustawienie nazwy działu, do którego przynależy sprzęt.

Parametry

<i>department</i>	Nowa nazwa dzialu sprzetu.
-------------------	----------------------------

4.1.3.5 setName()

```
void Equipment::setName (  
    string name )
```

Metoda publiczna umozliwiajaca ustawienie nazwy sprzetu.

Parametry

<i>name</i>	Nowa nazwa sprzetu.
-------------	---------------------

Dokumentacja dla tej klasy zostala wygenerowana z plikow:

- header.h
- dbfunction.cpp

4.2 Dokumentacja klasy EquipmentDatabase

Zaradza baza danych sprzetu.

```
#include <header.h>
```

Metody publiczne

- EquipmentDatabase ()
Konstruktor klasy EquipmentDatabase.
- vector< Equipment > getEquipmentList () const
Zwraca liste sprzetu w bazie danych.
- void addEquipment (Equipment equipment)
Dodaje sprzet do bazy danych.
- void removeEquipment (int id)
Usuwa sprzet z bazy danych na podstawie identyfikatora.
- void modifyEquipment (int id, string name, string department)
Modyfikuje dane sprzetu w bazie danych na podstawie identyfikatora.
- vector< Equipment > searchEquipment (string query) const
Wyszukuje sprzet w bazie danych na podstawie nazwy lub ID.
- vector< Equipment > getEquipmentByDepartment (string department) const
Zwraca sprzet z danej kategorii dzialu.

4.2.1 Opis szczegółowy

Zarządza baza danych sprzętu.

Klasa EquipmentDatabase służy do zarządzania bazą danych sprzętu. Klasa EquipmentDatabase ma prywatne pole equipment, które jest wektorem przechowującym obiekty klasy Equipment. Klasa ta ma również publiczne metody getEquipmentList(), addEquipment(), removeEquipment(), modifyEquipment(), searchEquipment() i getEquipmentByDepartment(), które pozwalają na wyświetlanie, dodawanie, usuwanie, modyfikowanie, wyszukiwanie i wyświetlanie sprzętu według działu.

4.2.2 Dokumentacja konstruktora i destruktor

4.2.2.1 EquipmentDatabase()

```
EquipmentDatabase::EquipmentDatabase ( )
```

Konstruktor klasy EquipmentDatabase.

4.2.3 Dokumentacja funkcji składowych

4.2.3.1 addEquipment()

```
void EquipmentDatabase::addEquipment (
    Equipment equipment )
```

Dodaje sprzęt do bazy danych.

4.2.3.2 getEquipmentByDepartment()

```
vector< Equipment > EquipmentDatabase::getEquipmentByDepartment (
    string department ) const
```

Zwraca sprzęt z danej kategorii działu.

Parametry

<i>department</i>	Nazwa działu, dla którego ma zostać zwrócony sprzęt.
-------------------	--

Zwraca

Lista sprzętu z danego działu.

4.2.3.3 getEquipmentList()

```
vector< Equipment > EquipmentDatabase::getEquipmentList ( ) const
```

Zwraca liste sprzętu w bazie danych.

4.2.3.4 modifyEquipment()

```
void EquipmentDatabase::modifyEquipment (
    int id,
    string name,
    string department )
```

Modyfikuje dane sprzętu w bazie danych na podstawie identyfikatora.

Parametry

<i>id</i>	Identyfikator sprzętu do modyfikacji.
<i>name</i>	Nowa nazwa sprzętu.
<i>department</i>	Nowa nazwa działu sprzętu.

4.2.3.5 removeEquipment()

```
void EquipmentDatabase::removeEquipment (
    int id )
```

Usuwa sprzęt z bazy danych na podstawie identyfikatora.

4.2.3.6 searchEquipment()

```
vector< Equipment > EquipmentDatabase::searchEquipment (
    string query ) const
```

Wyszukuje sprzęt w bazie danych na podstawie nazwy lub ID.

Parametry

<i>query</i>	Zapytanie wyszukiwania sprzętu.
--------------	---------------------------------

Zwraca

Lista sprzętu pasującego do zapytania.

Dokumentacja dla tej klasy została wygenerowana z plików:

- header.h
- dbfunction.cpp

Rozdział 5

Dokumentacja plików

5.1 Dokumentacja pliku dbfunction.cpp

```
#include "header.h"
```

Funkcje

- void displayEquipmentList (const EquipmentDatabase &db)
Wyswietla pelny wykaz sprzetu.
- bool loginAsAdmin ()
Funkcja loginAsAdmin loguje uzytkownika na konto Administratora.
- void addNewEquipment (EquipmentDatabase &db)
Funkcja addNewEquipment dodaje nowy sprzet do bazy danych db.
- void removeEquipment (EquipmentDatabase &db)
Funkcja removeEquipment usuwa sprzet z bazy danych db na podstawie jego ID.
- void modifyEquipment (EquipmentDatabase &db)
Funkcja modifyEquipment modyfikuje dane o sprzecie w bazie danych db na podstawie jego ID.
- void searchEquipment (EquipmentDatabase &db)
Funkcja searchEquipment wyszukuje sprzet w bazie danych db na podstawie podanego zapytania.
- void displayEquipmentByDepartment (EquipmentDatabase &db)
Funkcja displayEquipmentByDepartment wyswietla sprzet zwiazany z danym dzialem w firmie na podstawie bazy danych db.
- void dbcopy (ifstream wykaz, string outname)
- void logowanie (EquipmentDatabase db)
Funkcja logowanie obsluje proces logowania do systemu, wykorzystujac baze danych db.
- void option (EquipmentDatabase db)
Funkcja option wyswietla opcje do wyboru dla uzytkownika na podstawie bazy danych db.

5.1.1 Dokumentacja funkcji

5.1.1.1 addNewEquipment()

```
void addNewEquipment (
    EquipmentDatabase & db )
```

Funkcja `addNewEquipment` dodaje nowy sprzęt do bazy danych `db`.

5.1.1.2 dbcopy()

```
void dbcopy (
    ifstream wykaz,
    string outname )
```

5.1.1.3 displayEquipmentByDepartment()

```
void displayEquipmentByDepartment (
    EquipmentDatabase & db )
```

Funkcja `displayEquipmentByDepartment` wyświetla sprzęt związany z danym działem w firmie na podstawie bazy danych `db`.

5.1.1.4 displayEquipmentList()

```
void displayEquipmentList (
    const EquipmentDatabase & db )
```

Wyświetla pełny wykaz sprzętu.

Funkcja `displayEquipmentList` wyświetla pełny wykaz sprzętu z bazy danych `db`.

5.1.1.5 loginAsAdmin()

```
bool loginAsAdmin ( )
```

Funkcja `loginAsAdmin` loguje użytkownika na konto Administratora.

Zwraca

Warto jeśli logowanie powiodło się; warto jeśli logowanie nie powiodło się.

5.1.1.6 logowanie()

```
void logowanie (
    EquipmentDatabase db )
```

Funkcja `logowanie` obsługuje proces logowania do systemu, wykorzystując baze danych `db`.

5.1.1.7 modifyEquipment()

```
void modifyEquipment (
    EquipmentDatabase & db )
```

Funkcja `modifyEquipment` modyfikuje dane o sprzęcie w bazie danych `db` na podstawie jego ID.

5.1.1.8 option()

```
void option (
    EquipmentDatabase db )
```

Funkcja `option` wyświetla opcje do wyboru dla użytkownika na podstawie bazy danych `db`.

5.1.1.9 removeEquipment()

```
void removeEquipment (
    EquipmentDatabase & db )
```

Funkcja `removeEquipment` usuwa sprzęt z bazy danych `db` na podstawie jego ID.

5.1.1.10 searchEquipment()

```
void searchEquipment (
    EquipmentDatabase & db )
```

Funkcja `searchEquipment` wyszukuje sprzęt w bazie danych `db` na podstawie podanego zapytania.

5.2 Dokumentacja pliku header.h

```
#include <iostream>
#include <string>
#include <iomanip>
#include <vector>
#include <cstdlib>
#include <fstream>
```

Komponenty

- class Equipment
Klasa Equipment reprezentuje sprzęt. Ma prywatne pola id_, name_ i department_, które są dostępne za pomocą publicznych metod getId(), getName() i getDepartment(). Klasa Equipment ma również publiczne metody setName() i setDepartment(), które pozwalają na modyfikowanie nazwy i działu sprzętu.
- class EquipmentDatabase
Zarządza bazą danych sprzętu.

Funkcje

- void displayEquipmentList (const EquipmentDatabase &db)
Wyświetla pełny wykaz sprzętu.
- bool loginAsAdmin ()
Funkcja loginAsAdmin loguje użytkownika na konto Administratora.
- void addNewEquipment (EquipmentDatabase &db)
Funkcja addNewEquipment dodaje nowy sprzęt do bazy danych db.
- void removeEquipment (EquipmentDatabase &db)
Funkcja removeEquipment usuwa sprzęt z bazy danych db na podstawie jego ID.
- void modifyEquipment (EquipmentDatabase &db)
Funkcja modifyEquipment modyfikuje dane o sprzęcie w bazie danych db na podstawie jego ID.
- void searchEquipment (EquipmentDatabase &db)
Funkcja searchEquipment wyszukuje sprzęt w bazie danych db na podstawie podanego zapytania.
- void displayEquipmentByDepartment (EquipmentDatabase &db)
Funkcja displayEquipmentByDepartment wyświetla sprzęt związany z danym działem w firmie na podstawie bazy danych db.
- void logowanie (EquipmentDatabase db)
Funkcja logowanie obsługuje proces logowania do systemu, wykorzystując bazę danych db.
- void option (EquipmentDatabase db)
Funkcja option wyświetla opcje do wyboru dla użytkownika na podstawie bazy danych db.

5.2.1 Dokumentacja funkcji

5.2.1.1 addNewEquipment()

```
void addNewEquipment (  
    EquipmentDatabase & db )
```

Funkcja addNewEquipment dodaje nowy sprzęt do bazy danych db.

5.2.1.2 displayEquipmentByDepartment()

```
void displayEquipmentByDepartment (  
    EquipmentDatabase & db )
```

Funkcja displayEquipmentByDepartment wyświetla sprzęt związany z danym działem w firmie na podstawie bazy danych db.

5.2.1.3 displayEquipmentList()

```
void displayEquipmentList (
    const EquipmentDatabase & db )
```

Wyswietla pelny wykaz sprzetu.

Funkcja `displayEquipmentList` wyswietla pelny wykaz sprzetu z bazy danych `db`.

5.2.1.4 loginAsAdmin()

```
bool loginAsAdmin ( )
```

Funkcja `loginAsAdmin` loguje uzytkownika na konto Administratora.

Zwraca

Warto jesli logowanie powiodlo sie; warto jesli logowanie nie powiodlo sie.

5.2.1.5 logowanie()

```
void logowanie (
    EquipmentDatabase db )
```

Funkcja `logowanie` obsluguje proces logowania do systemu, wykorzystujac baze danych `db`.

5.2.1.6 modifyEquipment()

```
void modifyEquipment (
    EquipmentDatabase & db )
```

Funkcja `modifyEquipment` modyfikuje dane o sprzecie w bazie danych `db` na podstawie jego ID.

5.2.1.7 option()

```
void option (
    EquipmentDatabase db )
```

Funkcja `option` wyswietla opcje do wyboru dla uzytkownika na podstawie bazy danych `db`.

5.2.1.8 removeEquipment()

```
void removeEquipment (
    EquipmentDatabase & db )
```

Funkcja `removeEquipment` usuwa sprzęt z bazy danych `db` na podstawie jego ID.

5.2.1.9 searchEquipment()

```
void searchEquipment (
    EquipmentDatabase & db )
```

Funkcja `searchEquipment` wyszukuje sprzęt w bazie danych `db` na podstawie podanego zapytania.

5.3 header.h

Idź do dokumentacji tego pliku.

```
00001 #pragma once
00002 #include <iostream>
00003 #include <string>
00004 #include <iomanip>
00005 #include <vector>
00006 #include <cstdlib>
00007 #include <fstream>
00008 using namespace std;
00009
00018 class Equipment {
00019 public:
00026     Equipment(int id, string name, string department);
00027
00032     int getId() const;
00033
00038     string getName() const;
00039
00044     string getDepartment() const;
00045
00050     void setName(string name);
00051
00056     void setDepartment(string department);
00057 private:
00058     int id_;
00059     string name_;
00060     string department_;
00061 };
00062
00074 class EquipmentDatabase {
00075 public:
00079     EquipmentDatabase();
00080
00085     vector<Equipment> getEquipmentList() const;
00086
00091     void addEquipment(Equipment equipment);
00092
00097     void removeEquipment(int id);
00098
00105     void modifyEquipment(int id, string name, string department);
00106
00112     vector<Equipment> searchEquipment(string query) const;
00113
00119     vector<Equipment> getEquipmentByDepartment(string department) const;
00120 private:
00121     vector<Equipment> equipment_;
00122 };
00124 //void dbcopy(ifstream wykaz, string outname);
00125
00131 void displayEquipmentList(const EquipmentDatabase& db);
00132
00139 bool loginAsAdmin();
00140
```

```
00148 void addNewEquipment(EquipmentDatabase& db);
00149
00154 void removeEquipment(EquipmentDatabase& db);
00155
00160 void modifyEquipment(EquipmentDatabase & db);
00161
00165 void searchEquipment(EquipmentDatabase& db);
00166
00170 void displayEquipmentByDepartment(EquipmentDatabase& db);
00171
00172
00177 void logowanie(EquipmentDatabase db);
00179 //void dbcopy(istream wykaz, string outname);
00180
00185 void option(EquipmentDatabase db);
```

5.4 Dokumentacja pliku main.cpp

```
#include <iostream>
#include <string>
#include <iomanip>
#include <vector>
#include <cstdlib>
#include <fstream>
#include "header.h"
```

Funkcje

- int main ()

5.4.1 Dokumentacja funkcji

5.4.1.1 main()

```
int main ( )
```


Skorowidz

- addEquipment
 - EquipmentDatabase, 11
- addNewEquipment
 - dbfunction.cpp, 15
 - header.h, 18
- dbcopy
 - dbfunction.cpp, 16
- dbfunction.cpp, 15
 - addNewEquipment, 15
 - dbcopy, 16
 - displayEquipmentByDepartment, 16
 - displayEquipmentList, 16
 - loginAsAdmin, 16
 - logowanie, 16
 - modifyEquipment, 17
 - option, 17
 - removeEquipment, 17
 - searchEquipment, 17
- displayEquipmentByDepartment
 - dbfunction.cpp, 16
 - header.h, 18
- displayEquipmentList
 - dbfunction.cpp, 16
 - header.h, 18
- Equipment, 7
 - Equipment, 7
 - getDepartment, 9
 - getId, 9
 - getName, 9
 - setDepartment, 9
 - setName, 10
- EquipmentDatabase, 10
 - addEquipment, 11
 - EquipmentDatabase, 11
 - getEquipmentByDepartment, 11
 - getEquipmentList, 12
 - modifyEquipment, 12
 - removeEquipment, 12
 - searchEquipment, 12
- getDepartment
 - Equipment, 9
- getEquipmentByDepartment
 - EquipmentDatabase, 11
- getEquipmentList
 - EquipmentDatabase, 12
- getId
 - Equipment, 9
- getName
 - Equipment, 9
- header.h, 17
 - addNewEquipment, 18
 - displayEquipmentByDepartment, 18
 - displayEquipmentList, 18
 - loginAsAdmin, 19
 - logowanie, 19
 - modifyEquipment, 19
 - option, 19
 - removeEquipment, 19
 - searchEquipment, 20
- loginAsAdmin
 - dbfunction.cpp, 16
 - header.h, 19
- logowanie
 - dbfunction.cpp, 16
 - header.h, 19
- main
 - main.cpp, 21
- main.cpp, 21
 - main, 21
- modifyEquipment
 - dbfunction.cpp, 17
 - EquipmentDatabase, 12
 - header.h, 19
- option
 - dbfunction.cpp, 17
 - header.h, 19
- removeEquipment
 - dbfunction.cpp, 17
 - EquipmentDatabase, 12
 - header.h, 19
- searchEquipment
 - dbfunction.cpp, 17
 - EquipmentDatabase, 12
 - header.h, 20
- setDepartment
 - Equipment, 9
- setName
 - Equipment, 10