МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ

“БРЕСТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ”

**ИНТЕЛЕКТУАЛЬНЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ**

ОТЧЁТ

По лабораторной работе № \_\_

Выполнил:

Студент группы ИИ-22

Копанчук Евгений Романович

Проверил\_:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Брест – 2021

ЦЕЛЬ РАБОТЫ

1. Изучение перегрузки функций.

2. Программирование классов.

СПИСОК ЗАДАЧ

Перегрузка функций (задачи 5 а, б).

Индивидуальные задания.

**Ход работы**

ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

1. Изучить правила перегрузки функций.

2. Реализовать задания по перегрузке функций.

3. Реализовать разработанные классы на языке С++ в виде консольных приложений, обеспечивающих использование их объектов.

4\*. Повторить п.3, реализовав разработанные классы в виде оконных приложений.

**Программная реализация:**

|  |  |
| --- | --- |
| void EARN(int cost);  void EARN(double cost);  void EARN(double cost, int incr); | User.h |
| void User::EARN(int cost) { balance += cost; }  void User::EARN(double cost) { balance += cost; }  void User::EARN(double cost, int incr) { balance += cost \* incr; } | **User.cpp** |
| bool TO\_COMPARE();  bool TO\_COMPARE(char fst, char snd);  bool TO\_COMPARE(int fst, int snd); | **Sum.h** |
| bool Sum::TO\_COMPARE() { return a == b && b == c; }  bool Sum::TO\_COMPARE(char fst, char snd) {  if (fst == 'a' && snd == 'b') return a == b;  if (fst == 'a' && snd == 'c') return a == c;  if (fst == 'b' && snd == 'c') return b == c;  }  bool Sum::TO\_COMPARE(int fst, int snd) {  if (fst == 1 && snd == 2) return a == b;  if (fst == 1 && snd == 3) return a == c;  if (fst == 2 && snd == 3) return b == c;  } | **Sum.cpp** |
| #pragma once  #include <iostream>  #include <ctime>  using std::cout;  using std::endl;  using std::string;  using std::ostream;  class Text  {  private:  string title;  string text;  string ps;  protected:  int line = 100;  public:  Text();  Text(string title, string text, string ps);  Text(const Text& txt);  void TO\_SET\_TITLE(string title);  void TO\_SET\_TEXT(string text);  void TO\_SET\_PS(string ps);  void TO\_SHOW\_TEXT();  int COUNT();  int COUNT(char chr);  int COUNT(char chr, int ln);  }; | **Text.h** |
| #include "Text.h"  Text::Text() {  title = "Lorem, ipsum.";  text = "Lorem ipsum, ... , fuga cumque rerum eius officia ea!";  ps = "dolor sit amet";  }  Text::Text(string title, string text, string ps) {  this->title = title;  this->text = text;  this->ps = ps;  }  Text::Text(const Text& txt) {  this->title = txt.title;  this->text = txt.text;  this->ps = txt.ps;  }  void Text::TO\_SET\_TITLE(string title) { this->title = title; }  void Text::TO\_SET\_TEXT(string text) { this->text = text; }  void Text::TO\_SET\_PS(string ps) { this->ps = ps; }  void Text::TO\_SHOW\_TEXT() {  for (int i = 0; i < line / 2 - title.size() / 2; i++) cout << ' ';  cout << title;  for (int i = 0; i < 5; i++) cout << ' ';  for (int i = 0; i < text.size(); i++)  if (i % line == 0) cout << endl;  else cout << text[i];  cout << '\n';  for (int i = 0; i < line - 10 - ps.size(); i++) cout << ' ';  cout << ps << '\n';  }  int Text::COUNT() {  return title.size() + text.size() + ps.size();  }  int Text::COUNT(char chr) {  int resault = 0;  for (int i = 0; i < title.size(); i++)  if (title[i] == chr) resault++;  for (int i = 0; i < text.size(); i++)  if (text[i] == chr) resault++;  for (int i = 0; i < ps.size(); i++)  if (ps[i] == chr) resault++;  return resault;  }  int Text::COUNT(char chr, int ln) {  int resault = 0;  for (int i = ln \* line - 1; i < (ln + 1) \* line - 1 || text.size() == i; i++)  if (text[i] == chr) resault++;  return resault;  } | **Text.cpp** |

**Пример работы некоторых функций:**

Sum sum(2, 2, 3);

cout << sum.TO\_COMPARE() << sum.TO\_COMPARE('a', 'b') << sum.TO\_COMPARE(1, 2) << endl;

User man;

man.EARN(100); man.TO\_SHOW\_BALANCE();

man.EARN(213.423); man.TO\_SHOW\_BALANCE();

man.EARN(100, 2); man.TO\_SHOW\_BALANCE();

Text txt; txt.TO\_SHOW\_TEXT();

cout << txt.COUNT() << endl; cout << txt.COUNT('a') << endl; cout << txt.COUNT('a', 5) << endl;

****