

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Чувашского государственного университета имени И. Н. Ульянова»

Факультет информатики и вычислительной техники

Кафедра вычислительной техники

Курсовая работа  
по дисциплине «Объектно-ориентированное программирование»  
на тему:  
**«Модель экспертной системы»**

Выполнил:  
студент группы ИВТ-41-20  
Галкин Дмитрий Сергеевич  
Проверил:  
доцент Обломов И. А.  
Оценка: \_\_\_\_\_  
Дата: \_\_\_\_\_

## Введение

Цель работы: закрепление теоретических знаний, полученных по данному курсу, и приобретение практических навыков формализации поставленной задачи, создания и использования эффективных структур данных и алгоритмов.

Объектно-ориентированное программирование (ООП) – парадигма разработки программных систем, где программа представляется в виде совокупности объектов – экземпляров классов, а классы образуют иерархию наследования.

Существует пять основных компонентов ООП: объект, сообщение, класс, наследование и метод.

Существует 4 принципа ООП: инкапсуляция, абстракция, наследование и полиморфизм.

Инкапсуляция – предоставляет ограничение доступа к данным и возможностям их изменения. Позволяет пользователю не задумываться о сложности реализации программного компонента, а взаимодействовать с ним посредством методов. [1]

Абстракция – описывает выделение главных, наиболее значимых характеристик и наоборот – отбрасывание второстепенных, незначительных. [1]

Наследование – механизм, который позволяет описывать новый класс на основе существующего (родительского). При этом свойства и функциональность родительского класса заимствуются новым классом. [1]

Полиморфизм – возможность работать с несколькими типами так, будто это один и тот же тип. При этом поведение объектов будет разным в зависимости от типа, к которому они принадлежат. [1]

## **Структура экспертной системы**

Экспертная система — компьютерная программа, которая, как и любая другая программа, получает какие-то данные на вход и формирует другие данные на выходе. Особенность в том, что целью работы экспертной системы является замена эксперта (человека) в некоторой предметной области.

Важнейшей частью экспертной системы являются базы знаний как модели поведения экспертов в определённой области знаний с использованием процедур логического вывода и принятия решений, иными словами, базы знаний — совокупность фактов и правил логического вывода в выбранной предметной области деятельности.

Закономерности для предсказания, прогнозирования основываются на данных. Данные — это совокупность фактов и идей, представленных в формализованном виде. Продвинутое интеллектуальные системы способны учиться на основе этих данных, добавляя новые знания в базу знаний. [2]

## **Постановка задачи**

В соответствии с заданием необходимо разработать программу, основанную на модели экспертной системы. Основным функционалом которой будет являться постановка и вывод вопросов пользователю, в зависимости от ответов выводить определенный результат. Если данная проблема не будет существовать в базе знаний эксперта, то будет добавлена возможность внесение новых данных в базу знаний.

## **Выбор метода, способа решения поставленной задачи и обоснование выбора**

В поисках метода решения поставленной задачи, было выявлено 2 типа вывода: обратный и прямой логический вывод.

**Прямая цепочка рассуждений (ПЦР)** – Суть метода прямой цепочки рассуждений заключается в изобретении вопросов, позволяющих на каждом шаге отбросить большое количество возможных ответов, так что правильный ответ может быть установлен быстро. Причем задаваемые при каждой проверке вопросы целиком зависят от возможных ответов. Различные ответы (т.е если бы были задуманы разные животные) подразумевают необходимость использования разных проверок. [3]

**Обратная цепочка рассуждений (ОЦР)** – Данную цепочку рассуждений, можно рассматривать и как продукционную модель вывода, так и модель логического программирования. Структура такой версии с обратной цепочкой рассуждений проектируется с помощью введений группы правил высокого уровня. Каждое такое правило описывает одно правило, четко указывая, какая именно информация нужна системе, чтобы прийти к выводу, что именно данное конкретное правило является искомым ответом. [3]

Мной был выбран вариант обратной цепочки рассуждений, т.к. я изначально интуитивно начал проектировать систему так, чтобы программа пришла к правильному ответу только после того, как получит ответы на все вопросы и сможет вывести результат исходя из всей картины.

Также я рассмотрел варианты технологий, через которые можно реализовать поставленную задачу.

**Первый способ** – создание Windows Forms на Visual Studio и подключение базы данных PostgreSQL. Преимущество данного способа заключается в том, что можно отрисовать красивое диалоговое окно чата, где происходило бы общение пользователя с экспертом. А также база знаний была бы полностью перенесена в базу данных, посредством которой можно было бы получить доступ как локально, так и дистанционно. Но для такой реализации нужно дополнительно скачивать графические библиотеки, изучение различных операторов и функций этих библиотек и основную сложность реализации было бы подключение базы данных к языку программирования C++, а также написание методов для получения нужных данных из базы. Это заняло бы большую часть времени написания курсовой работы. Используя все функции графических библиотек и методы базы данных, позволило бы разработать полноценное приложение с приятным интерфейсом и более гибким способом получения данных из вне, хранящихся не напрямую в проекте приложения.

**Второй способ** – вывод общения эксперта с пользователем в консольном приложении CLion значительно упрощает и ускоряет время работы над курсовым проектом. Для этого необходимо в базовом классе KnowLedgeBase (база знаний нашей экспертной системы) определить такие поля, как dictQuestion – словарь с ключом «наименование проблемы» и со значением «истинности ответа на вопрос», и структурой problems – где хранится проблема и список вопросов, наводящих к данной проблеме. Также методы поиска и добавление проблемы в базе знаний. Производный от класса KnowLedgeBase класс Expert (выступает экспертом, который общается с пользователем) должен иметь поля в виде итератора для обращения к вопросу в словаре базе знаний с методами вывода результата, поиска проблемы и печати вопросов пользователю. Недостаток этого метода в том, что процесс общения происходит не в диалоговом экране, а в консоли и данные хранятся локально в проекте на ПК.

Мной был выбран второй способ, так как основная цель данной курсовой работы, которая была указана ранее — это закрепление знаний и навыков по ООП, которое подразумевает наследование, имеющееся во втором способе.

## Теоретический раздел

В качестве предметной области нашей экспертной системы была выбрана «техническое обслуживание ПК»

Класс **KnowLedgeBase** выступает в качестве базы знаний для эксперта. Уже изначально хранит в себе словарь исходных вопросов и ответы на них в качестве **false**.

```
map<string, bool> dictQuestions = {"Отсутствует реакция на нажатие кнопки включения? ", false},
    {"Все кабели питания подключены правильно? ", false},
    {"Провода и коннекторы внутри ПК исправны? ", false},
    {"При отсоединении кнопки перезагрузки от материнской платы начинает
работать? ", false},
    {"При замыкании двух контактов с надписью 'Power Switch' происходит запуск
компьютера? ", false},
    {"Включенный в сеть кабель блок питания не начинает работать при замыкании
черного и зеленого контактов? ", false},
    {"При нажатии кнопки запуска компьютер начинает работать? ", false},
    {"Компьютер сразу прекращает работу? ", false}};
```

При помощи конструктора будут добавляться проблемы и список наводящих вопросов к данной проблеме.

Пример одного из них

```
public:
    KnowledgeBase() {
        problem pr = {"Неисправна кнопка перезагрузки", {"Отсутствует реакция на нажатие кнопки
включения? ", "Все кабели питания подключены правильно? ", "Провода и коннекторы внутри ПК
исправны? ", "При отсоединении кнопки перезагрузки от материнской платы начинает работать?
"};
        problems.push_back(pr);
    }
```

Класс **Expert** является посредником, который будет вытягивать данные из базы знаний и задавать вопросы пользователю, полученные ответы будут отправлять обратно в базу знаний.

Метод *askQuestion* и будет задавать вопросы пользователю и отправлять данные в базу знаний для изменения параметров ответа в вышеупомянутой переменной *dictQuestions*.

```
void askAQuestion() {
    iter = dictQuestions.begin();
    string str;
    while (iter != dictQuestions.end()) {
        cout << iter->first << endl << "Ответ: ";
        cin >> str;
    }
```

```

if (strstr(str.c_str(),"Да") || strstr(str.c_str(),"да")) {
    iter->second = true;
}
else if (strstr(str.c_str(),"Нет") || strstr(str.c_str(),"нет")) {
    iter->second = false;
}

iter.operator++();
}

showAnswer();
}

```

Ниже приведу полное графическое описание схемы базы знаний нашей экспертной системы, при котором будут выдаваться результаты в зависимости от ответа пользователя.

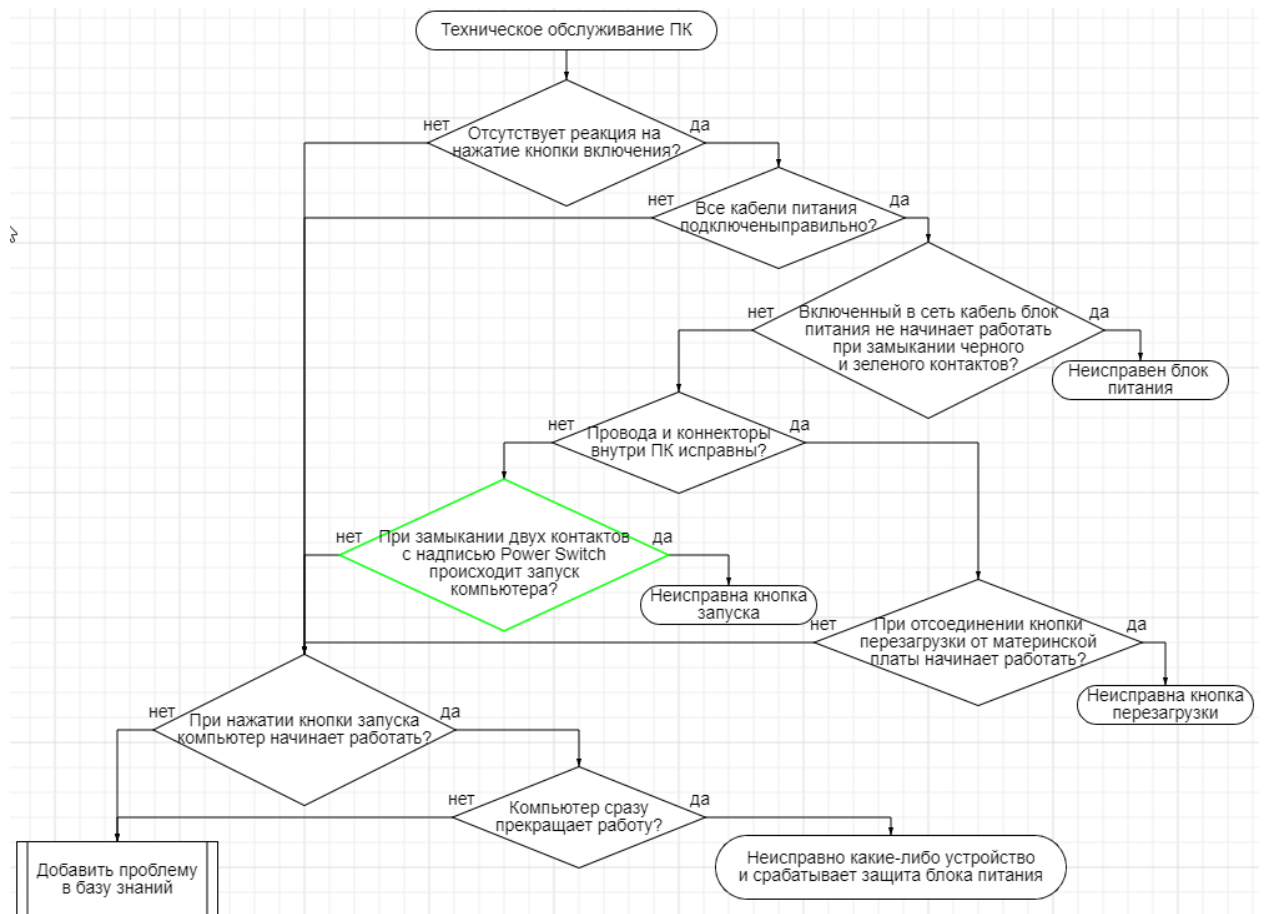


Рисунок 1. Модель ЭС

Для вывода проблемы в консоль реализован метод *showAnswer*, здесь происходит анализ всех ответов пользователь при помощи метода *findAnswer*, который обращается к методу поиска из базового класса.

```

void showAnswer() {
    string str;
    int answer = findAnswer();
}

```



```

cout << "Процесс выявления проблемы завершен. " << endl;
if (answer == -1) {
    cout << "По полученным ответам в моей базе знаний не существует подобной проблемы\n
Хотели бы вы добавить новую проблему в мою базу знаний? " << endl << "Answer: ";
    cin >> str;

    if (strstr(str.c_str()), "Да") || strstr(str.c_str()), "да")) { addProblem(); }
} else {
    cout << "Вашей проблемой может являться: " << problems.at(answer).name << endl;
}

cout << "Спасибо, что воспользовались нашей системой обслуживания. Хорошего дня:)" << endl;
}

```

- Если после анализа эксперт смог найти подходящую проблему в базе знаний, то он её выводит
- Если после анализа эксперт не смог выявить проблему, то он предлагает внести новую проблему в базу знаний с помощью метода базового класса *addProblem*. Пользователь вводит новую проблему и список наводящих вопросов к ней.

```

-----
|   Вас приветствует экспертная система технического обслуживания ПК   |
-----
- Для выявления проблемы нужно будет ответить на мои вопросы.
- Либо да
- Либо нет
Включенный в сеть кабель блок питания не начинает работать при замыкании черного и зеленого контактов?
Ответ: Да
Все кабели питания подключены правильно?
Ответ: |

```

Рисунок 2. Процесс ответа на вопросы эксперта

На всех этапах работы экспертной системы реализован консольный дружественный интерфейс, чтобы у пользователя возникало меньше вопросов о происходящей работе системы, и он быстро понимал, что требуется вести в данный момент с клавиатуры.

Пример приветственного диалогового окна:

```

void show_start() {
    cout << "-----" << endl;
    cout << "|   Вас приветствует экспертная система технического обслуживания ПК   |" << endl;
    cout << "-----" << endl;
    cout << "- Для выявления проблемы нужно будет ответить на мои вопросы." << endl;
    cout << "- Либо да" << endl;
    cout << "- Либо нет" << endl;
}

```

## **Выводы по работе**

В ходе данной курсовой работы была написана программа, моделирующая экспертную систему на примере технического обслуживания ПК.

Использование объектно-ориентированного программирования удобно, а в ряде случаев практически необходимо в создании больших программных продуктов.

Итогом курсовой работы можно считать закрепление знаний в области создания алгоритмов навыков программирования на языке высокого уровня C++, полученных в ходе обучения по курсу «Объектно-ориентированное программирование».

### **Список использованной литературы**

1. <https://javarush.ru/groups/posts/1966-principih-obhhektno-orientirovannogo-programmirovanija> (дата обращения 22.05.2022)
2. <https://habr.com/ru/post/346236/> (дата обращения 22.05.2022)
3. Марселлус Д. Программирование экспертных систем на турбо прологе, 1994 (дата обращения 28.05.2022)

## Оглавление

Введение .....	2
Структура экспертной системы .....	3
Постановка задачи .....	4
Выбор метода, способа решения поставленной задачи и обоснование выбора .....	5
Теоретический раздел.....	7
Выводы по работе .....	10
Список использованной литературы .....	11
Приложение.....	13

## Приложение

### Текст программы

#### main.cpp

```
#include <iostream>
#include "Headers/Expert.h"

using namespace std;

/*****Prototype func*****/
void show_start(); // show main text
/*****/

/*
 * Start expert system
 */
int main () {
    show_start();
    Expert expert;
    expert.askAQuestion();

    return 0;
}

/*****Func*****/
void show_start() {
    cout << "-----" << endl;
    cout << "|   Вас приветствует экспертная система технического обслуживания ПК   |" << endl;
    cout << "-----" << endl;
    cout << "- Для выявления проблемы нужно будет ответить на мои вопросы." << endl;
    cout << "- Либо да" << endl;
    cout << "- Либо нет" << endl;
}
/*****/
```

#### KnowLedgeBase.h

```
//
// Created by dimaa on 08.05.2022.
//

#ifndef CHUVSU_2KURS_OOP_KNOWLEDGEBASE_H
#define CHUVSU_2KURS_OOP_KNOWLEDGEBASE_H

#include <map>
#include <iostream>
#include <cstring>
```

```

#include <string>
#include <cstdlib>
#include <vector>
using namespace std;

struct problem {
    string name;
    vector<string> questions;
};

class KnowledgeBase {
private:
    int countProblems = 4;
protected:
    map<string, bool> dictQuestions = {{"Отсутствует реакция на нажатие кнопки включения? ",
false},
{"Все кабели питания подключены правильно? ", false},
{"Провода и коннекторы внутри ПК исправны? ", false},
{"При отсоединении кнопки перезагрузки от материнской платы начинает
работать? ", false},
{"При замыкании двух контактов с надписью 'Power Switch' происходит
запуск компьютера? ", false},
{"Включенный в сеть кабель блок питания не начинает работать при
замыкании черного и зеленого контактов? ", false},
{"При нажатии кнопки запуска компьютер начинает работать? ", false},
{"Компьютер сразу прекращает работу? ", false}};
    vector<problem> problems;

    bool findAnswerInProblems(int index) {
        problem pr = problems.at(index);
        string question;
        for (int i = 0; i < pr.questions.size(); ++i) {
            question = pr.questions.at(i);
            if (!dictQuestions[question]) {
                return false;
            }
        }

        return true;
    }

    void addProblem() {

        cout << "- Процесс внесения данных в базу знаний!" << endl;
        cout << "Добавьте наименования проблемы: ";
        cin >> pr.name;

        cout << "Добавьте список вопросов (после окончания добавления вопросов напишите 'end')"
```

```

<< endl;
    string str;
    while (true) {
        cin >> str;
        if (strstr(str.c_str(), "end")) {
            break;
        }
        pr.questions.push_back(str);
    }

    cout << "Процесс внесения новых данных в базу знаний закончен!" << endl;
}

public:
    KnowledgeBase() {
        problem pr = {"Неисправна кнопка перезагрузки", {"Отсутствует реакция на нажатие кнопки
включения? ",
                                                                "Все кабели питания подключены правильно? ",
                                                                "Провода и коннекторы внутри ПК исправны? ",
                                                                "При отсоединении кнопки перезагрузки от материнской
платы начинает работать? "}};
        problems.push_back(pr);

        pr = {"Неисправна кнопка запуска", {"Отсутствует реакция на нажатие кнопки включения? ",
                                                                "Все кабели питания подключены правильно? ",
                                                                "Провода и коннекторы внутри ПК исправны? ",
                                                                "При замыкании двух контактов с надписью 'Power Switch'
происходит запуск компьютера? "}};
        problems.push_back(pr);

        pr = {"Неисправен блок питания", {"Отсутствует реакция на нажатие кнопки включения? ",
                                                                "Все кабели питания подключены правильно? ",
                                                                "Включенный в сеть кабель блок питания не начинает работать
при замыкании черного и зеленого контактов? "}};
        problems.push_back(pr);

        pr = {"Неисправно какое-либо устройство и срабатывает защита блока питания", {"При
нажатии кнопки запуска компьютер начинает работать? ",
                                                                "Компьютер сразу прекращает работу? "}};
        problems.push_back(pr);
    }

};

#endif //CHUVSU_2KURS_OOP_KNOWLEDGEBASE_H

```

## Expert.h

```
//  
// Created by dimaa on 08.05.2022.  
//  
  
#ifndef CHUVSU_2KURS_OOP_EXPERT_H  
#define CHUVSU_2KURS_OOP_EXPERT_H  
  
#include <map>  
#include <iostream>  
#include <cstring>  
#include <cstdlib>  
#include "KnowledgeBase.h"  
  
using namespace std;  
  
class Expert : public KnowledgeBase{  
private:  
    int findAnswer() {  
        for (int i = 0; i < problems.size(); ++i) {  
            if (findAnswerInProblems(i)) {  
                return i;  
            }  
        }  
        return -1;  
    }  
  
    void showAnswer() {  
        string str;  
        int answer = findAnswer();  
        cout << "Процесс выявления проблемы завершен. " << endl;  
        if (answer == -1) {  
            cout << "По полученным ответам в моей базе знаний не существует подобной  
проблемы\nХотели бы вы добавить новую проблему в мою базу знаний? " << endl << "Answer: ";  
            cin >> str;  
  
            if (strstr(str.c_str()), "Да") || strstr(str.c_str()), "да") { addProblem(); }  
        } else {  
            cout << "Вашей проблемой может являться: " << problems.at(answer).name << endl;  
        }  
  
        cout << "Спасибо, что воспользовались нашей системой обслуживания. Хорошего дня:) " <<  
endl;  
    }  
  
public:  
    Expert() = default;  
    map<string, bool>::iterator iter;
```



```

void askAQuestion() {
    iter = dictQuestions.begin();
    string str;
    while (iter != dictQuestions.end()) {
        cout << iter->first << endl << "Ответ: ";
        cin >> str;

        if (strstr(str.c_str(), "Да") || strstr(str.c_str(), "да")) {
            iter->second = true;
        }
        else if (strstr(str.c_str(), "Нет") || strstr(str.c_str(), "нет")) {
            iter->second = false;
        }

        iter.operator++();
    }

    showAnswer();
}

};

#endif //CHUVSU_2KURS_OOP_EXPERT_H

```

Примеры выполнения программы:

Попробуем определить проблему «Неисправна кнопка перезагрузки»:

```
-----
|   Вас приветствует экспертная система технического обслуживания ПК   |
|-----|
- Для выявления проблемы нужно будет ответить на мои вопросы.
- Либо да
- Либо нет
Включенный в сеть кабель блок питания не начинает работать при замыкании черного и зеленого контактов?
Ответ: Нет
Все кабели питания подключены правильно?
Ответ: Да
Компьютер сразу прекращает работу?
Ответ: Нет
Отсутствует реакция на нажатие кнопки включения?
Ответ: Да
При замыкании двух контактов с надписью 'Power Switch' происходит запуск компьютера?
Ответ: Нет
При нажатии кнопки запуска компьютер начинает работать?
Ответ: Нет
При отсоединении кнопки перезагрузки от материнской платы начинает работать?
Ответ: Да
Провода и коннекторы внутри ПК исправны?
Ответ: Да
Процесс выявления проблемы завершен.
Вашей проблемой может являться: Неисправна кнопка перезагрузки
Спасибо, что воспользовались нашей системой обслуживания. Хорошего дня:)
```

Рисунок 3. Скриншот из программы CLion

Попробуем определить «Неисправно какое-либо устройство и срабатывает защита блока питания» (пошагово):

```
-----
|   Вас приветствует экспертная система технического обслуживания ПК   |
|-----|
- Для выявления проблемы нужно будет ответить на мои вопросы.
- Либо да
- Либо нет
Включенный в сеть кабель блок питания не начинает работать при замыкании черного и зеленого контактов?
Ответ: Нет
Все кабели питания подключены правильно?
Ответ: Да
Компьютер сразу прекращает работу?
Ответ: Да
Отсутствует реакция на нажатие кнопки включения?
Ответ: Нет
При замыкании двух контактов с надписью 'Power Switch' происходит запуск компьютера?
Ответ: Нет
При нажатии кнопки запуска компьютер начинает работать?
Ответ: Да
При отсоединении кнопки перезагрузки от материнской платы начинает работать?
Ответ: Нет
Провода и коннекторы внутри ПК исправны?
Ответ: Да
Процесс выявления проблемы завершен.
Вашей проблемой может являться: Неисправно какое-либо устройство и срабатывает защита блока питания
Спасибо, что воспользовались нашей системой обслуживания. Хорошего дня:)
```

**Приветственное сообщение**

**Шаг 1**

**Шаг 2**

**Шаг 3**

**Шаг 4**

**Шаг 5**

**Шаг 6**

**Шаг 7**

**Шаг 8**

**Результат**

Рисунок 4. Скриншот из программы CLion

Попробуем определить проблему, которой нет в базе знаний:

```
-----
|   Вас приветствует экспертная система технического обслуживания ПК   |
|-----|
- Для выявления проблемы нужно будет ответить на мои вопросы.
- Либо да
- Либо нет
Включенный в сеть кабель блок питания не начинает работать при замыкании черного и зеленого контактов?
Ответ: Нет
Все кабели питания подключены правильно?
Ответ: Нет
Компьютер сразу прекращает работу?
Ответ: Нет
Отсутствует реакция на нажатие кнопки включения?
Ответ: Нет
При замыкании двух контактов с надписью 'Power Switch' происходит запуск компьютера?
Ответ: Нет
При нажатии кнопки запуска компьютер начинает работать?
Ответ: Нет
При отсоединении кнопки перезагрузки от материнской платы начинает работать?
Ответ: Нет
Провода и коннекторы внутри ПК исправны?
Ответ: Нет
Процесс выявления проблемы завершен.
По полученным ответам в моей базе знаний не существует подобной проблемы
Хотели бы вы добавить новую проблему в мою базу знаний?
Ответ: Да
- Процесс внесения данных в базу знаний!
Добавьте наименования проблемы: Неисправен системный блок
Добавьте список вопросов (после окончания добавления вопросов напишите 'end')
Компьютер не запускается
Неисправен блок питания
end
Процесс внесения новых данных в базу знаний закончен!
Спасибо, что воспользовались нашей системой обслуживания. Хорошего дня:)
```

Рисунок 5. Скриншот из программы CLion