

ПРОГРАММНАЯ РЕАЛИЗАЦИЯ ЭКСПЕРТНОЙ СИСТЕМЫ СТАНОЧНОГО МОДУЛЯ

Дмитров В.В.

*Саратовский государственный технический университет имени Гагарина Ю.А.,
Саратов*

Ключевые слова: экспертная система, диагностика, ремонт, поиск дефектов.

Аннотация. Цель исследования – разработать программу для экспертной системы станочного модуля. В процессе разработки был разработан алгоритм для экспертной системы, написана и протестирована программа для диагностики станка и поиска дефектов. Программа разработана на языке программирования C++. В программу так же встроен графический интерфейс для удобного взаимодействия с пользователем. В современном мире, разработка и исследование различных вариантов экспертных систем в диагностике станочных модулей, это самое перспективное развитие для производства, а также является экономически, научной и практически выгодным направлением современности.

SOFTWARE IMPLEMENTATION OF THE EXPERT SYSTEM OF THE MACHINE MODULE

Dmitrov V.V.

Saratov State Technical University named after Gagarin Yu.A., Saratov

Keywords: expert system, diagnostics, repair, search for defects.

Abstract. The purpose of the study is to develop a program for the expert system of the machine module. During the development process, an algorithm for the expert system was developed, a program for machine diagnostics and defect detection was written and tested. The program is developed in the C++ programming language. The program also has a built-in graphical interface for convenient user interaction. In the modern world, the development and research of various variants of expert systems in the diagnosis of machine modules is the most promising development for production, and is also an economically, scientifically and practically profitable direction of our time.

Модель этой программы – система поддержки принятия решений, призвана выбрать нужное решение, и тем самым облегчить этот процесс в определенной ситуации. Данные системы могут использоваться в машиностроении, экономике, образовании, медицине, сельском хозяйстве и других областях.

Основной код программы написан на C++ и Qt:C++, так как я принял решение использовать Qt – фреймворк. Позволяет разработать графическое приложение для операционных систем с помощью конструктивных технологий.

Алгоритмы получения решений при поиске отказавшего блока. Одним из главных моментов экспертной системы поддержки принятия решений является исследование алгоритмов самой работы экспертной системы в определённых условиях и методов принятия ответов. Алгоритм - это ядро программной части экспертной системы [1].

Идея метода парных сравнений состоялась в том, чтобы попарно сравнивались каждые два объекта и определялось первенство одного из них, отсюда это название.

Интерфейс данной экспертной системы разрешил выполнять взаимодействие между оператором (пользователем) и экспертной системой, для обмена информацией. Он прост в использовании и восприятии для оператора [2].

Процедурная компонента содержит правила, используемые для преобразования декларативной информации, представленной продукционными правилами вида: если *предпосылка*, то *заключение*.

Сущность – значение – это пара, которая представляет факты: предпосылку и заключение. Они содержатся в базе данных декларативной компоненты. Процедурная часть имеет информацию касающихся процедур принятия решений [3]. Другим языком, заготовленное специальное значение, вызываемое при характерном ему состоянии объекта диагностики, будет соответствовать специальному ответу экспертной системы. Ответ может содержать до нескольких специальных значений (сущность-значений). База знаний может содержать любое число основных правил, в зависимости от требуемой точности. Если подсистема АСМ выдала отказ при диагностике, то начинается проверка полученных данных с эталонными данными, согласно с (1).

$$z_{in} \leq z_i \leq z_{ie} \tag{1}$$

где z_{in} и z_{ie} – нижнее и верхнее допустимые значения параметра

Мною построен алгоритм последовательности диагностирования основных подсистем на основании критерии качества.

Определил применяемые виды сбора информации от экспертов с помощью интервью, анкет и дискуссий, метод “мозгового штурма”, метод Дельфы. После, анализировал методы математической обработки информации, полученной от экспертов, выбирал метод парных сравнений, т.к. он прост в реализации и даёт возможность определить и откорректировать причинно-следственные связи.

Пример реализации части алгоритма, представлен на рисунке 1.

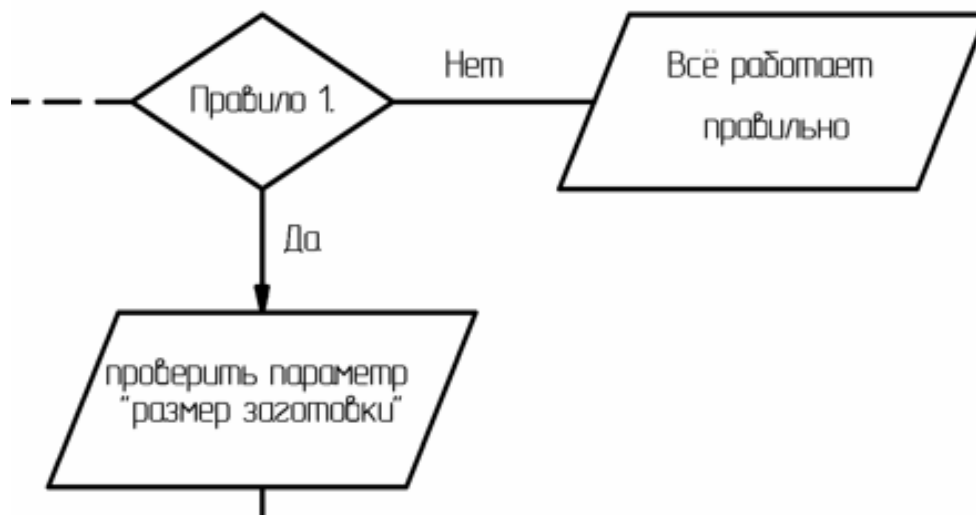


Рис. 1. Параметр качества “размер (x, z)” => отклонение от нормы

Мною была разработана программа, на основе изученных данных и примеров.

На рисунке 2 представлен пример рекомендации, по критериям “Размерность” и “Шероховатость”.

Частичный код листинг экспертной системы представлен на рисунках 3, 4.

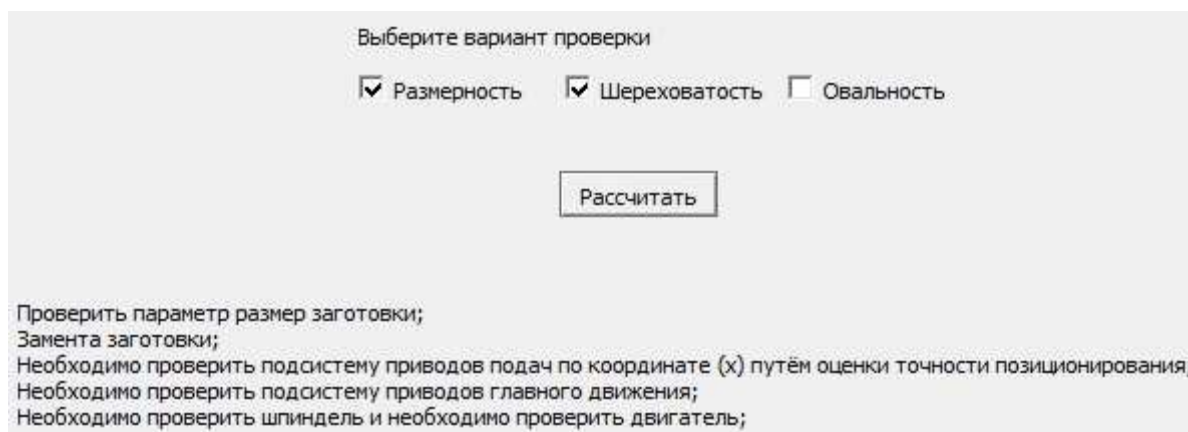


Рис. 2. Наглядный пример вывода программы

```

1  #ifndef SECONDDWINDOW_H
2  #define SECONDDWINDOW_H
3  #include <QWidget>
4  #include <QDialog>
5  #include "thirdwindow.h"
6
7  namespace Ui {
8  class SecondWindow;
9  }
10
11 class SecondWindow : public QDialog
12 {
13     Q_OBJECT
14
15 public:
16     explicit SecondWindow(QWidget *parent = nullptr);
17     ~SecondWindow();
18
19 private slots:
20     void on_pushButton_clicked();
21
22     void on_pushButton_2_clicked();
23
24 private:
25     Ui::SecondWindow *ui;
26     thirdwindow *window_second;
27 };
28
29 #endif // SECONDDWINDOW_H
30

```

Рис. 3. Первое окно программы

```

1  #ifndef MAINWINDOW_H
2  #define MAINWINDOW_H
3
4  #include <QMainWindow>
5  #include <secondwindow.h>
6
7  QT_BEGIN_NAMESPACE
8  namespace Ui { class MainWindow; }
9  QT_END_NAMESPACE
10
11 class MainWindow : public QMainWindow
12 {
13     Q_OBJECT
14
15 public:
16     MainWindow(QWidget *parent = nullptr);
17     ~MainWindow();
18
19 private slots:
20     void on_pushButton_clicked();
21
22 private:
23     Ui::MainWindow *ui;
24     SecondWindow *window_first;
25 };
26 #endif // MAINWINDOW_H
27

```

Рис. 4. Второе окно программы

На основе проведенного исследования я выявляю достоинства подобного способа выполнения программы: быструю реализацию интерфейса, гибкость элементов интерфейса, быструю настройку, возможность мульти платформенной реализации, а также в Qt используется более низкоуровневый подход: здесь точкой входа является C++ код, который уже будет создавать окно программы.

Особое внимание уделялось разработке экспертной системы поддержки принятия решений при диагностировании токарного станочного модуля на базе Qt с использованием языка C++ с целью обеспечения качества выпускаемых изделий и решены поставленные задачи.

Применение фреймворка Qt ускорило работу программы по сравнению со сборками на Electron'е и имеющимися аналогами, что позволит реализовывать функцию диагностирования в реальном времени производственного процесса. Использование модульной структуры программы позволит в процессе эксплуатации облегчить работу специалиста по обновлению базы знаний.

Список литературы

1. Игнатъев А.А., Козлова Т.Д., Самойлова Е.М. Модель базы знаний экспертной системы поддержки процесса диагностирования автоматических станочных модулей // Известия высших учебных заведений. Поволжский регион. Технические науки. – 2014. – №2(30). – С. 16-23. – URL: https://izvuz_tn_eng.pnzgu.ru/files/izvuz_tn_eng.pnzgu.ru/2214.pdf.
2. Самойлова Е.М., Мусатов В.Ю. Цифровая трансформация проектирования и управления автоматизированных и автоматических производств: Учебное пособие. – Саратов: Ай Пи Ар Медиа, 2019. – 160 с.
3. Бржозовский Б.М., Игнатъев А.А., Мартынов В.В., Схиртладзе А.Г. Учебник для вузов «Диагностика и надежность автоматизированных систем» / под ред. Б.М. Бржозовского. – Саратов: Саратовский государственный технический университет, 2007. – 352 с.
4. Игнатъев А.А., Самойлова Е.М., Игнатъев С.А. Интеллектуальные технологии в машиностроении: Учебное пособие. – Саратов: Саратовский государственный технический университет, 2012 – 95 с.
5. Игнатъев А.А., Козлова Т.Д., Самойлова Е.М. Экспертная система поддержки процесса диагностирования автоматических станочных модулей: монография. – Саратов: Изд-во Саратов. гос. техн. ун-та, 2014. – 101 с.

Сведения об авторе:

Дмитров Владислав Владимирович – студент.