ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»

Факультет компьютерных наук
Департамент программной инженерии

СОГЛАСОВАНО
Доцент, старший научный сотрудник
научно-учебной лаборатории ПОИС,
к.ф.-м.н.

_____ А.А. Каленкова
«___» _____ 2018 г.

УТВЕРЖДАЮ
Академический руководитель
образовательной программы
«Программная инженерия»

_____ В.В. Шилов
«___» _____ 2018 г.

Подп. и дата	
Инв. № дубл.	
Взам. инв. №	
Подп. и дата	
Инв. № подл	RU.17701729.507140-01 34 01-1

ПРОГРАММА ПОСТРОЕНИЯ РЕГУЛЯРНОГО ВЫРАЖЕНИЯ ПО КОНЕЧНОМУ АВТОМАТУ

Руководство оператора

ЛИСТ УТВЕРЖДЕНИЯ

RU.17701729.04.13-01 34 01-1-ЛУ

Исполнитель:
студент группы БПИ173
_____ /Таранцова Полина Дмитриевна /
«___»_____ 2018 г.

2018

УТВЕРЖДЕНО
RU.17701729.04.13-01 34 01-1-ЛУ

ПРОГРАММА ПОСТРОЕНИЯ РЕГУЛЯРНОГО ВЫРАЖЕНИЯ ПО КОНЕЧНОМУ АВТОМАТУ

Руководство оператора

RU.17701729.04.13-01 34 01-1

Листов 16

Инв. № подл	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата
RU.17701729.507140-01 34 01-1				

2018

СОДЕРЖАНИЕ

1. Назначение программы.....3
1.1 Функциональное назначение 3
1.2 Эксплуатационное назначение.....3
2. Условия выполнения программы 3
2.1 Минимальный состав аппаратных средств 3
2.2 Минимальный состав программных средств 3
2.3 Требования к персоналу (пользователю) 3
3. Выполнение программы 4
3.1 Установка программы 4
3.2 Запуск программы и работа с приложением.....4
ПРИЛОЖЕНИЕ 14
ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ.....16

1. Назначение программы

1.1 Функциональное назначение

Функциональным назначением программы является работа с загруженным в программу из внешних источников или построенным пользователем конечным автоматом, предполагающая последовательное приведение конечного автомата к соответствующему регулярному выражению посредством метода удаления состояний. В качестве выходных данных программа предоставляет регулярное выражение, полученное с применением алгоритма построения регулярного выражения с заменой подпроцессов и без него, а также анализ эффективности данного алгоритма.

1.2 Эксплуатационное назначение

Возможности программы предполагают ее использование в научных лабораториях для анализа работы алгоритмов, анализа логов событий в рамках теории глубинного анализа процессов, а также в образовательных учреждениях для ознакомления студентов с теорией автоматов и формирования у учащихся навыков построения конечных автоматов, их минимизации и приведения к регулярному выражению. Клиентская часть программы выполняется на персональных компьютерах.

Предполагается, что данный продукт будет использоваться в научных лабораториях для анализа работы алгоритмов, а также в образовательных учреждениях для ознакомления студентов с теорией автоматов и формирования у учащихся навыков построения конечных автоматов, их минимизации и приведения к регулярному выражению.

2. Условия выполнения программы

2.1 Минимальный состав аппаратных средств

- процессор с частотой 2,2 Ггц или выше;
- 0.5 гигабайт (ГБ) (для 32-разрядной системы) или 1 ГБ (для 64-разрядной системы) оперативной памяти (ОЗУ);
- 0.5 гигабайт (ГБ) (для 32-разрядной системы) или 1 ГБ (для 64-разрядной системы) пространства на жестком диске;
- монитор с разрешением 1024×640 или более высоким;
- клавиатура и мышь.

2.2 Минимальный состав программных средств

- операционная система Windows 7 или более поздняя версия;
- установленный Microsoft .NET Framework версии не ниже 4.0.

2.3 Требования к персоналу (пользователю)

Пользователь должен обладать базовыми знаниями теории автоматов и моделей процессов.

3. Выполнение программы

В данном разделе описан пример работы с программой.

3.1 Установка программы

Для установки программы необходимо скопировать папку «Expression Miner» с информационного накопителя на компьютер.

3.2 Запуск программы и работа с приложением

Приложение запускается при открытии файла «Project.exe». После запуска появляется стартовое окно приложения.

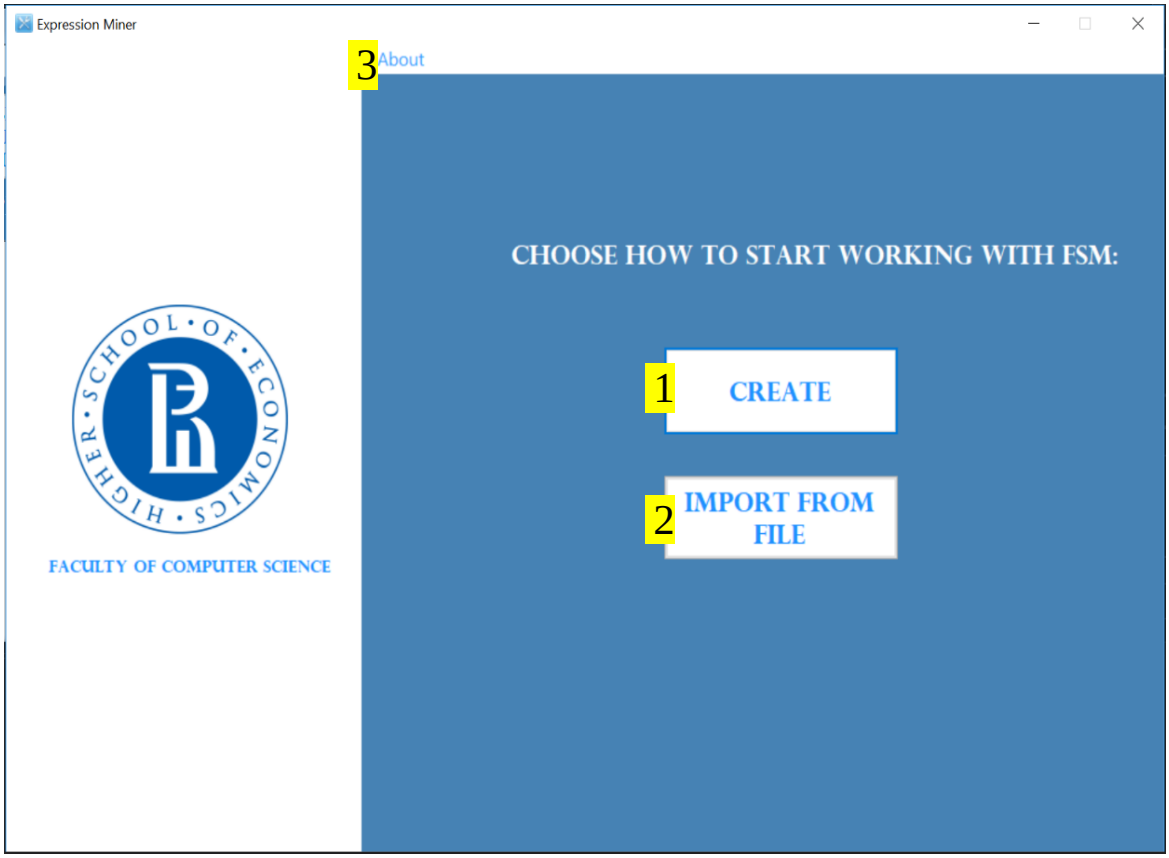


Рис. 1. Стартовое окно приложения.

После начала работы программы Вам будет предложено выбрать, в каком формате предоставить программе конечный автомат для ее работы.

Для создания конечного автомата в программе нажмите 1.

Для загрузки в программу конечного автомата в файле формата .tsml нажмите 2.

Для получения информации о разработчике нажмите 3.

Построение автомата в программе

После нажатия кнопки 1 на стартовом окне (Рис. 1) появится окно с инструментами для построения конечного автомата.

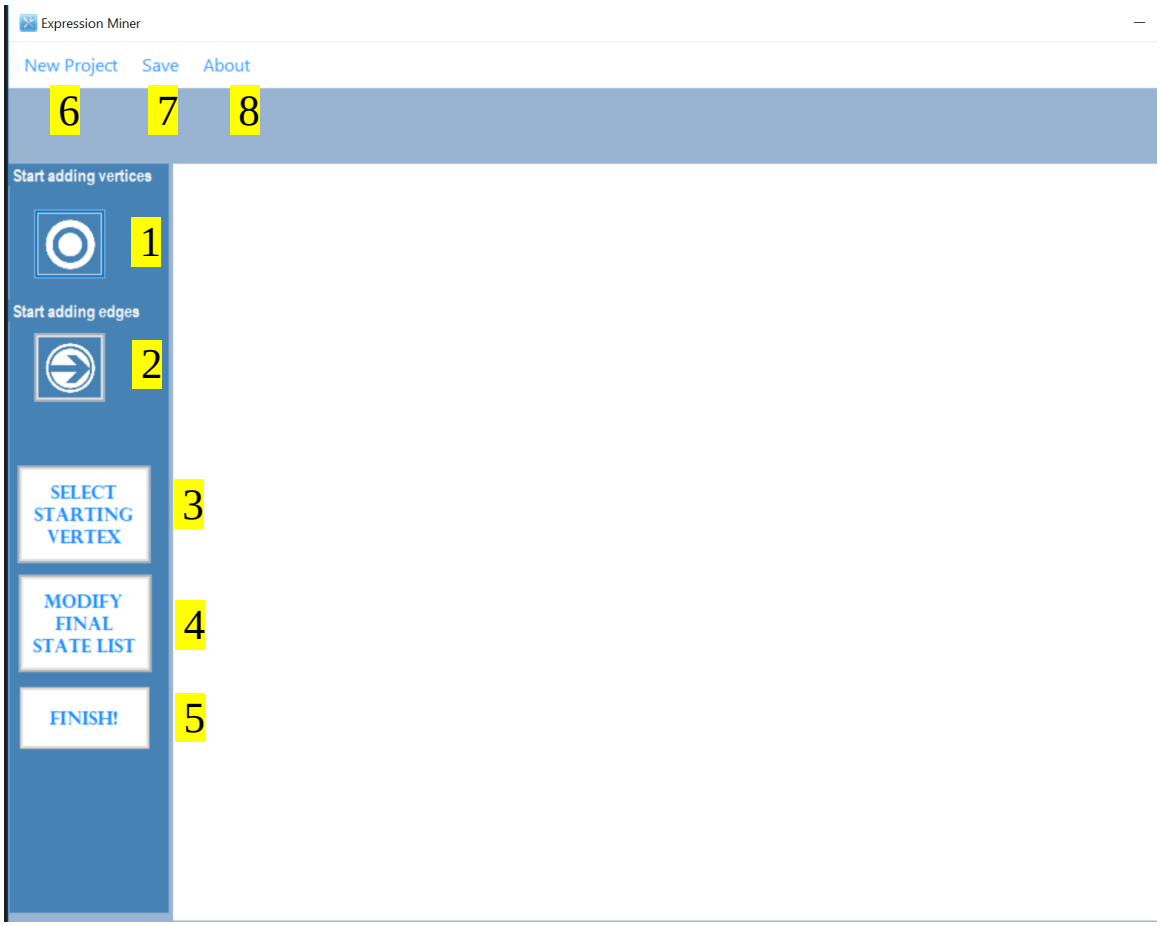


Рис. 2. Окно с инструментами для построения конечного автомата.

Для включения режима добавления состояния нажмите 1.

Для включения режима добавления перехода нажмите 2.

Для включения режима выбора начального состояния нажмите 3.

Для включения режима редактирования списка допускающих состояний нажмите 4.

Для завершения построения автомата и перехода к работе с ним нажмите 5.

Включенный режим выключается нажатием на кнопку включения этого режима, также сбрасывается нажатием на кнопку включения любого другого режима.

Для начала нового проекта нажмите 6.

Для сохранения построенного автомата нажмите 7. Однако нельзя сохранить автомат, не имеющий начального и допускающего состояний.

Для получения информации о разработчике нажмите 8.

Добавление состояния

Нажмите на кнопку 1 (Рис. 1), затем нажмите на поле отрисовки на месторасположение состояния.

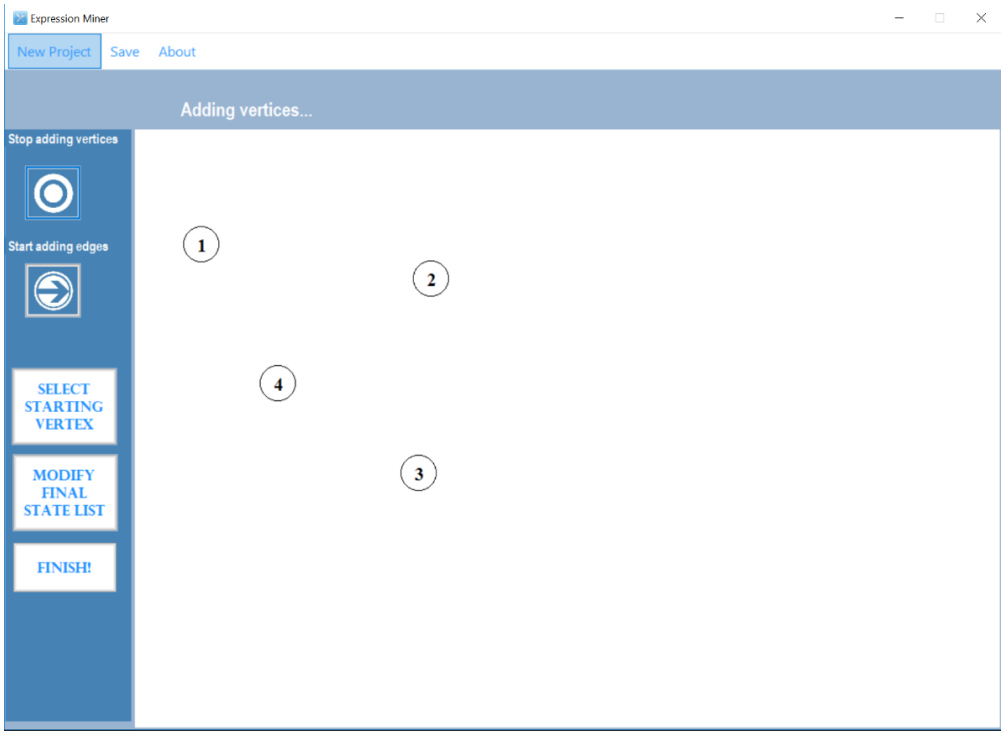


Рис. 3. Добавление состояний.

Добавление перехода

Нажмите на кнопку 2 (Рис. 2). Нажмите на состояние исхода.

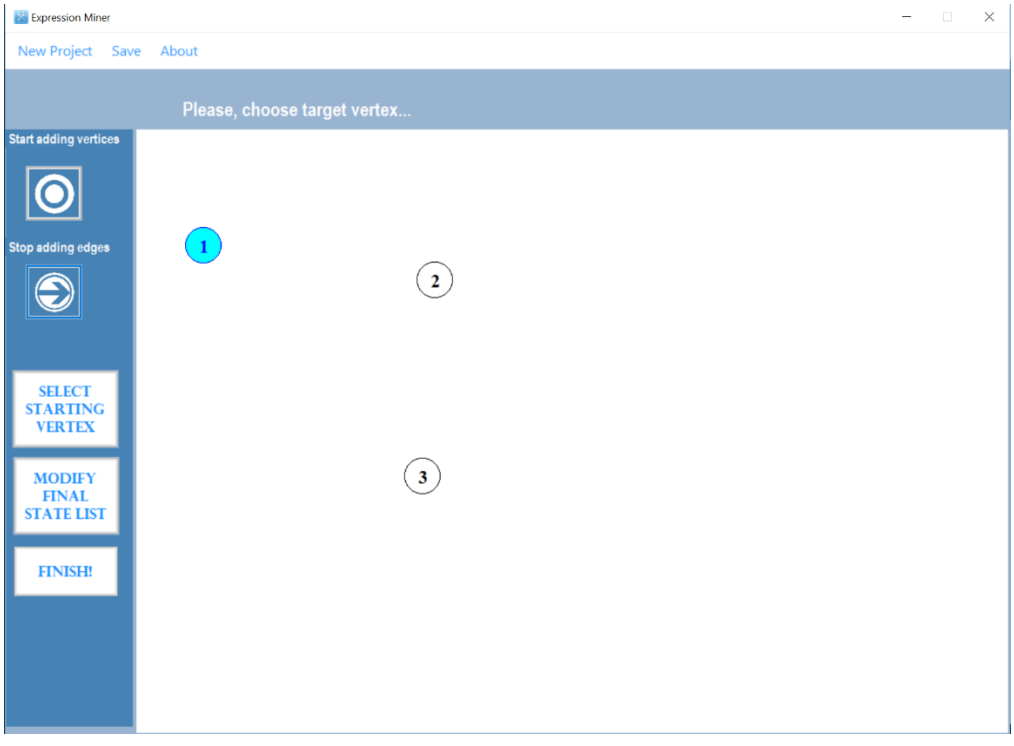


Рис. 4. Выбор состояния исхода.

Нажмите на состояние входа. Введите во всплывающем окне регулярное выражение, задающее переход.

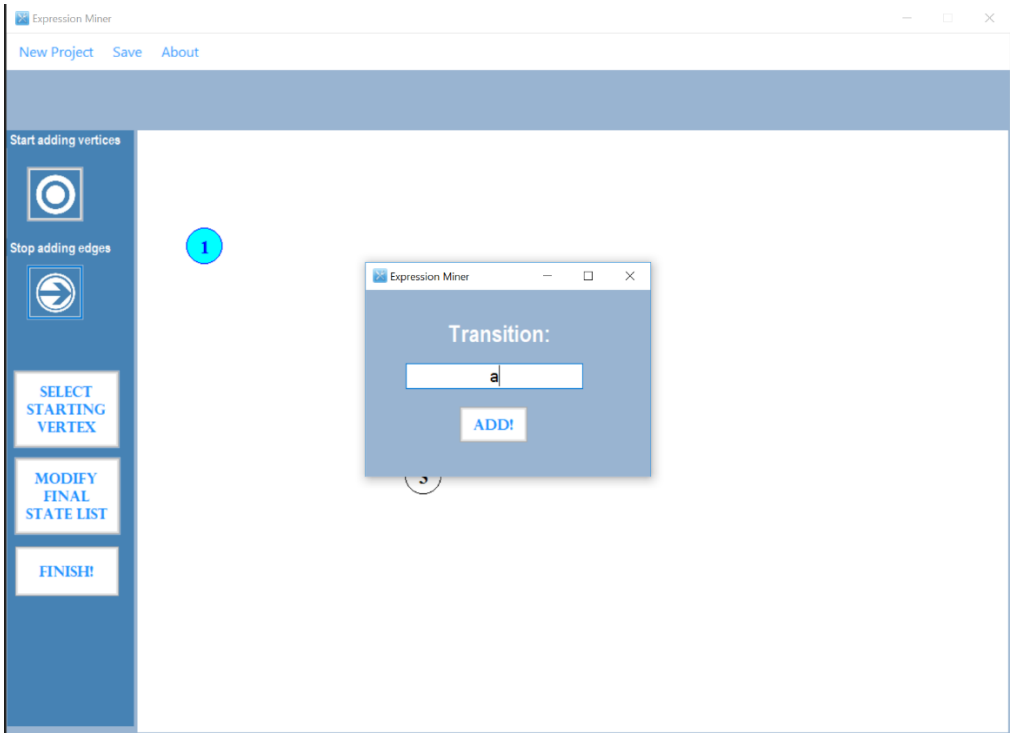


Рис. 5. Ввод регулярного выражения, задающего переход.

Нажмите «Add», чтобы добавить переход.

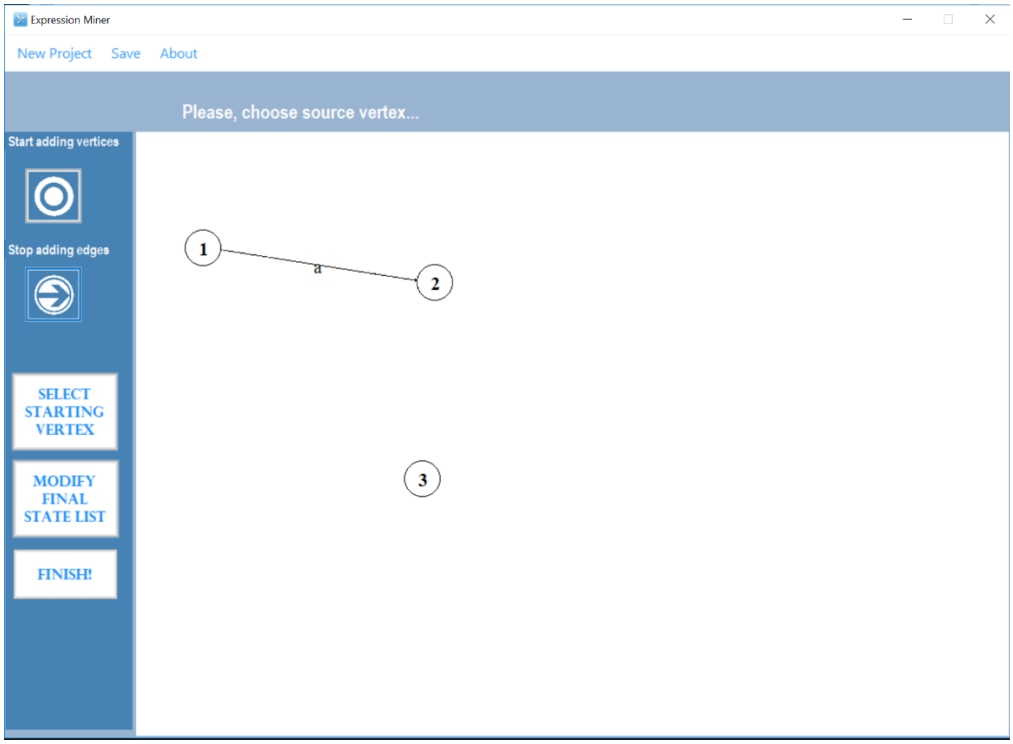


Рис. 6. Добавлен переход.

Удаление состояний и переходов

Для удаления состояния выберите удаляемое состояние нажатием на него правой кнопкой мыши. Затем подтвердите удаление состояния.

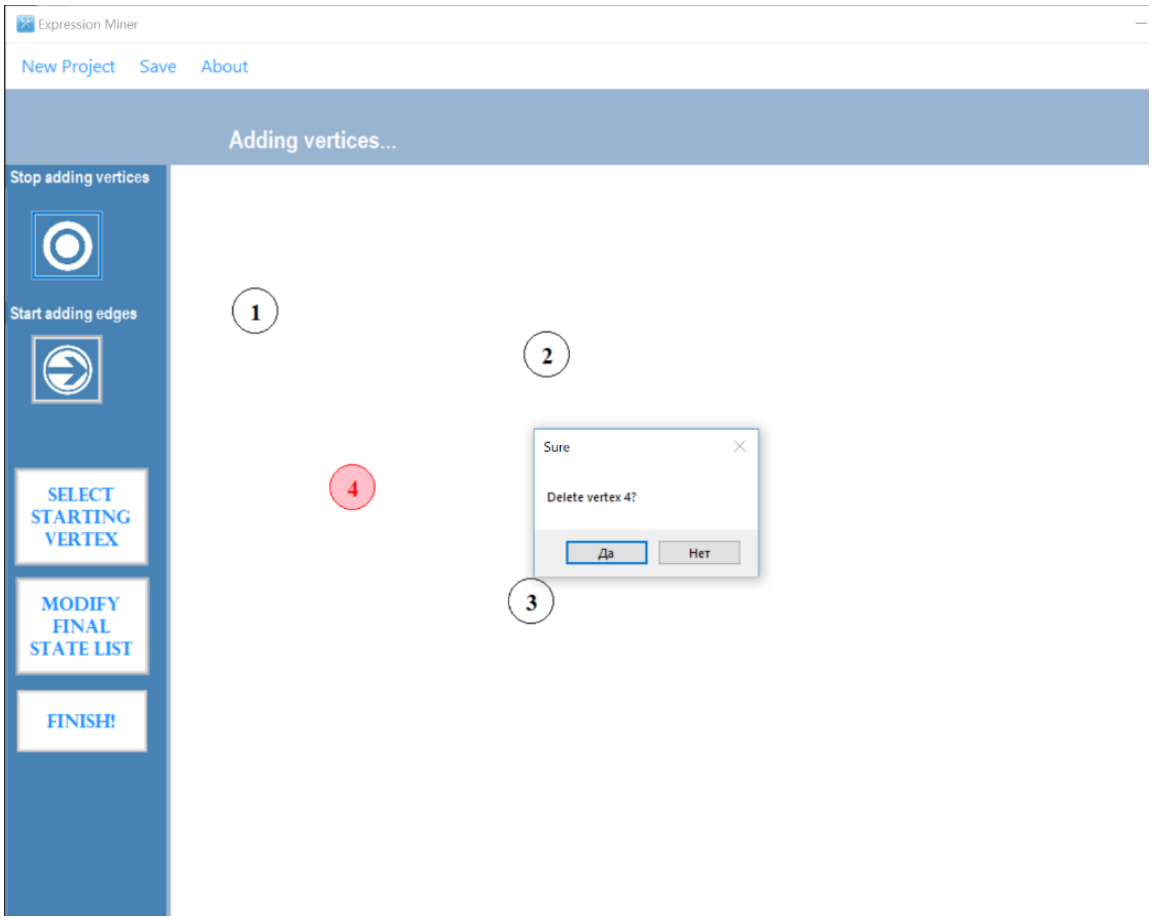


Рис. 7. Удаление состояния.

Для удаления перехода выберите удаляемый переход нажатием на него правой кнопкой мыши. Затем подтвердите удаление перехода.

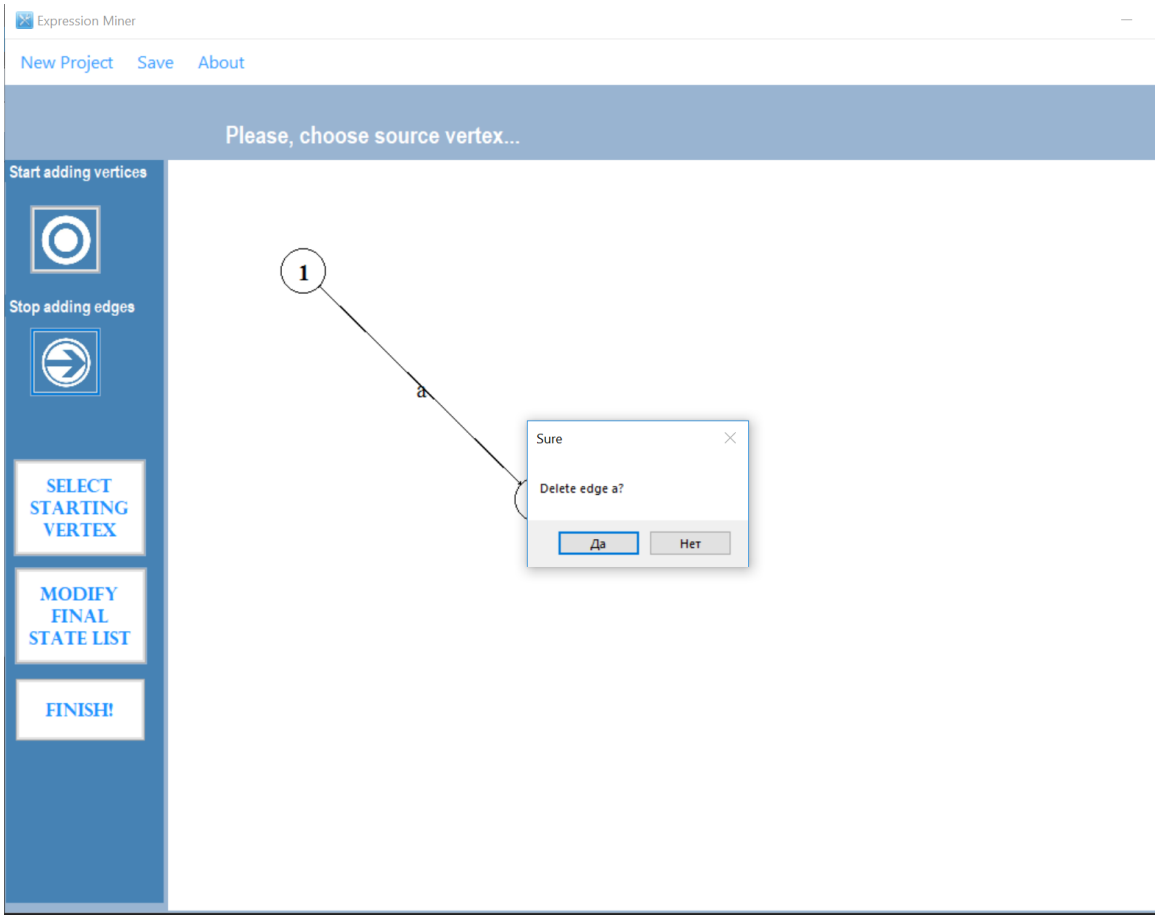


Рис. 8. Удаление переходов.

Построение регулярного выражения по конечному автомату

Данное окно открывается после нажатия кнопки 5 на рис. 2 или загрузки конечного автомата из файла (кнопка 1 на рис. 1).

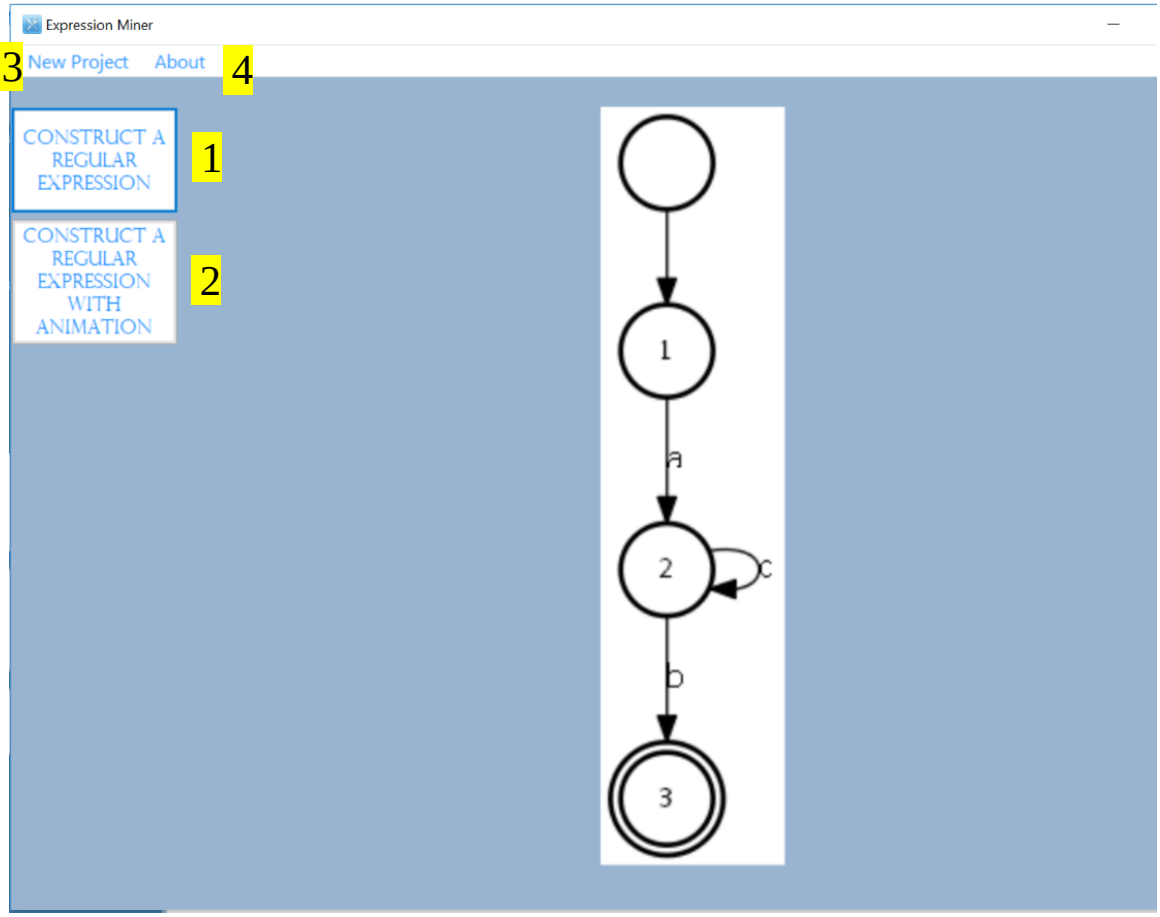


Рис. 9. Окно выбора способа построения регулярного выражения.

Для построения регулярного выражения без анимации нажмите 1.

Для построения регулярного выражения с анимацией нажмите 2.

Для начала нового проекта нажмите 3.

Для получения информации о разработчике нажмите 4.

Построение регулярного выражения с анимацией

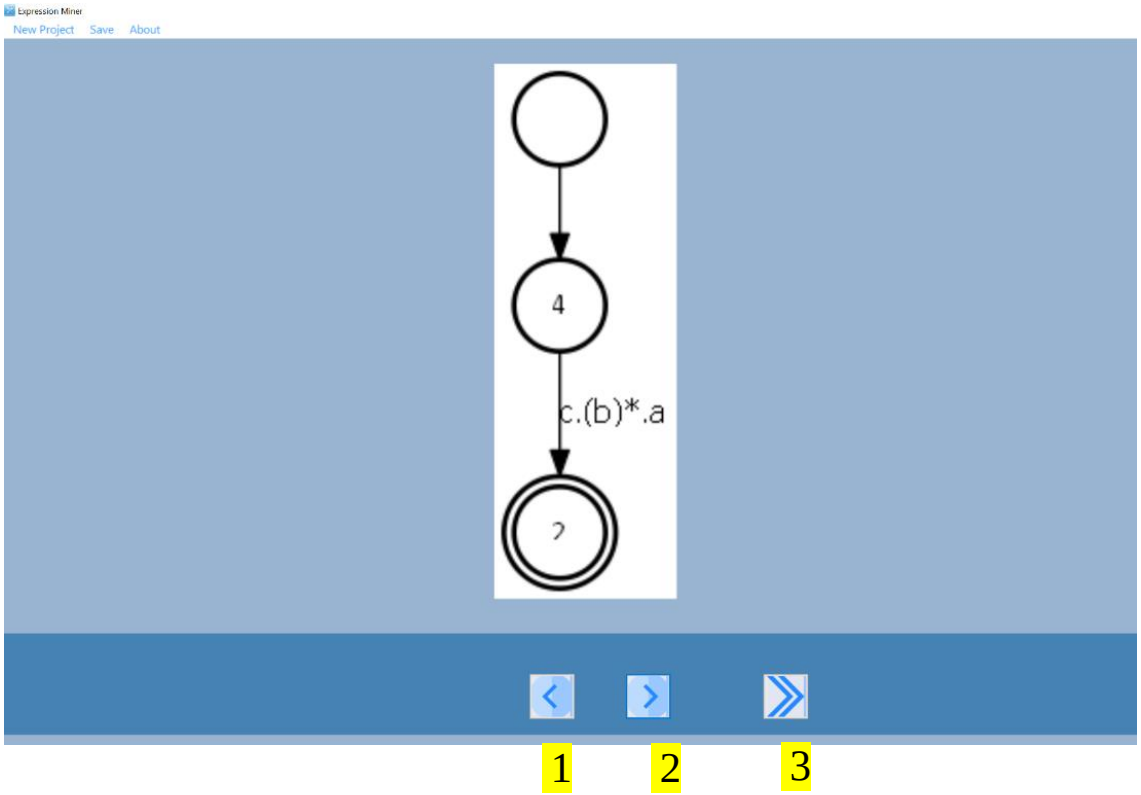


Рис. 10. Окно построения регулярного выражения с анимацией.

Чтобы вернуться к предыдущему шагу анимации, нажмите 1.

Чтобы перейти к следующему шагу анимации, нажмите 2.

Чтобы пропустить все шаги анимации, начиная с текущего, нажмите 3.

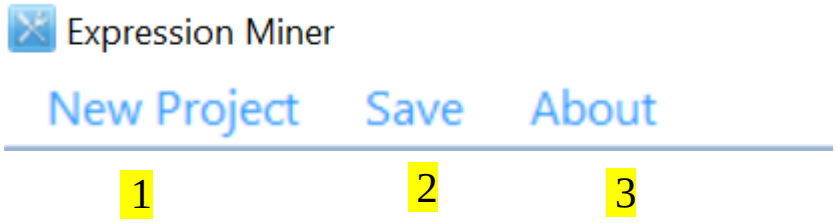


Рис. 11. Верхнее меню.

Для начала нового проекта нажмите 1.

Для сохранения текущего шага анимации нажмите 2.

Для получения информации о разработчике нажмите 3.

Результат работы программы

Данное окно открывается после нажатия кнопки 1 на Рис. 9 или кнопки 3 на Рис. 10. Оно содержит построенное регулярное выражение и аналитические данные.

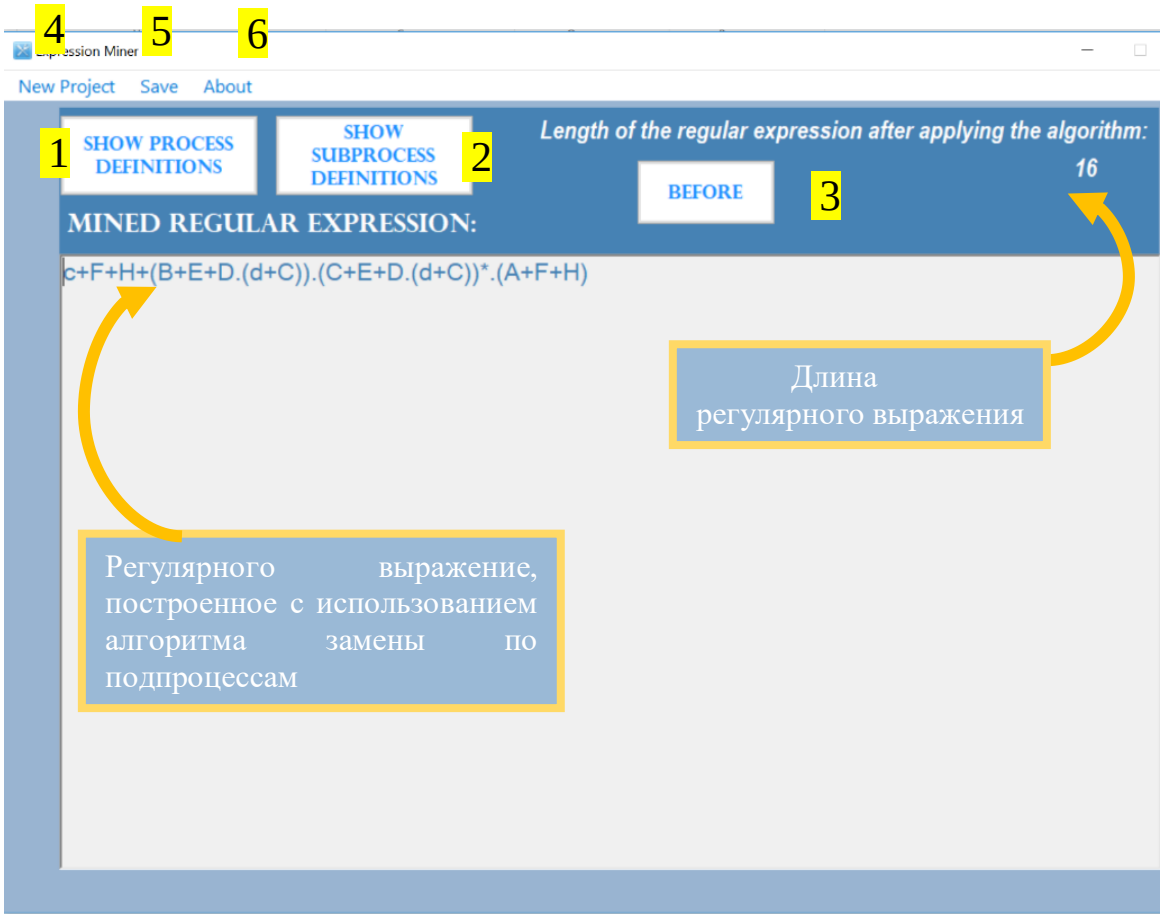


Рис. 12. Окно с построенным регулярным выражением.

Для открытия словаря процессов нажмите 1.

Для открытия словаря подпроцессов нажмите 2.

Для открытия окна с регулярным выражением, построенным без применения алгоритма замены по подпроцесса.

Для создания нового проекта нажмите 4.

Для сохранения результата работы программы нажмите 5.

Для получения информации о разработчике нажмите 6.

Сохранение результата работы программы

После нажатия кнопки 5 (Рис. 12) во всплывающем окне с параметрами сохранения выберите данные, которые необходимо сохранить в файл (1 – Рис. 13).

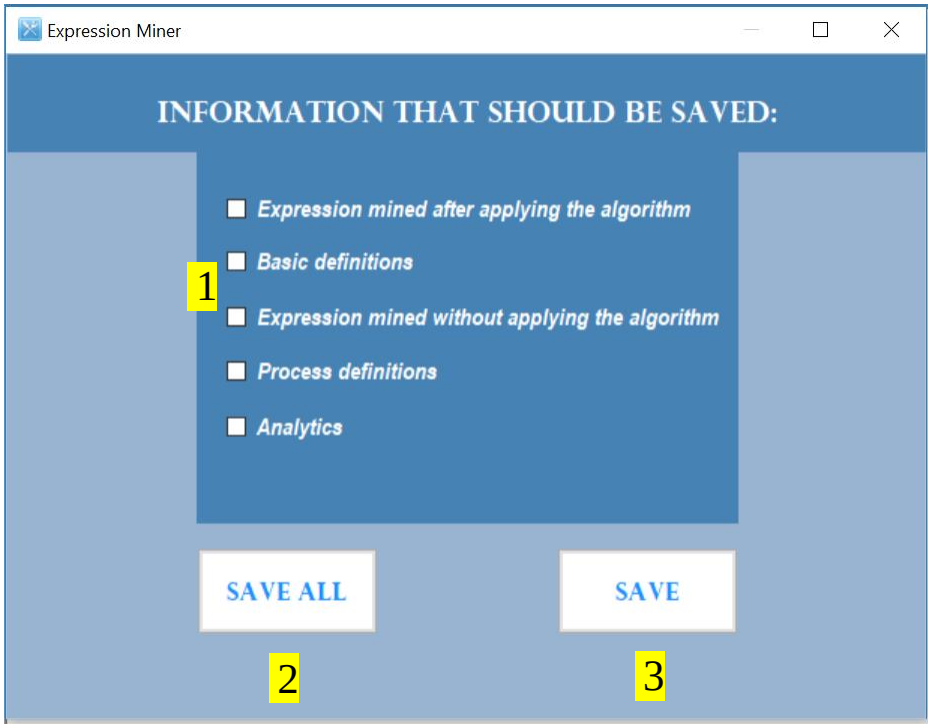


Рис. 13. Параметры сохранения.

Чтобы сохранить все полученные данные, нажмите 2.

Чтобы сохранить только выбранные данные, нажмите 3.

ПРИЛОЖЕНИЕ

ТЕРМИНОЛОГИЯ

Таблица 1.1

Термин (рус)	Термин (англ)	Определение
Глубинный анализ процессов	**Process mining**	Область исследований, занимающаяся построением моделей процессов по логам событий.
Способ представления	**Representational bias**	Набор ограничений, налагаемых на модели целевых процессов, играет важную роль глубинном анализе процессов.
Регулярные выражения	**Regular expressions**	Способ моделирования последовательных процессов, использующий шаблоны выбора, повторения и итерации.
Метод удаления состояний	**State removal method**	Алгоритм построения регулярного выражения по конечному автомату, предполагающий последовательное удаление состояний автомата и одновременное построение регулярных выражений, соответствующих заменяемым системам переходов.
Язык	**Language**	Множество последовательностей действий.
Регулярный язык	**Regular language**	Регулярный язык – это язык на множестве активностей $\mathcal{A}$, если он может быть построен индуктивно, опираясь на следующие правила: 1. $\emptyset$ - это регулярный язык; 2. $\{\langle\,\rangle\}$ – это регулярный язык; 3. $\{\langle a_i\rangle\}$, где $a_i \in \mathcal{A}$ – это регулярный язык, обозначаемый a_i; 4. Пусть $P, Q \subseteq \mathcal{A}^*$ - регулярные выражения, тогда: 4.1. $P.Q = \{\langle p_1, \dots, p_n, q_1, \dots, q_m\rangle \mid \langle p_1, \dots, p_n\rangle \in P \mid \langle q_1, \dots, q_m\rangle \in Q\}$ – также является регулярным выражением; 4.2. $P \mid Q = \{\sigma \mid \sigma \in P \cup Q\}$ - также является регулярным выражением; 4.3. $P^* = U_{i=0}^{\infty} P^n$, где $P^0 = \{\langle\,\rangle\}, P^1 = P$, и $P^n = P^{n-1}.P$ при $n > 0$ - также является регулярным выражением.

ТЕРМИНОЛОГИЯ

Таблица 1.2

Термин (рус)	Термин (англ)	Определение
Конечный автомат	**Finite state machine (FSM)**	Конечный автомат – это кортеж $FSM = (S, E, T, s_i, S_f)$, где S – конечное множество состояний, E– конечное множество событий, $T \subseteq (S \times E \times S)$ – конечное множество переходов, $s_i \subseteq S$ – начальное состояние, и $S_f \subseteq S$ – множество конечных состояний.
Начальное состояние	**Initial state**	Стартовое состояние автомата.
Конечные (допускающие) состояния	**Accepting states**	Состояния, в которых автомат может завершить работу.
Событие (Процесс)	**Event**	Действие, способное изменить состояние автомата.
Переход	**Transition**	Событие, связывающее два состояния автомата.
Длина регулярного выражений	**Length of the regular expression**	Количество символов без учета ключевых знаков.
Максимальная вложенность	**Maximum nesting**	Максимальное количество замен символов переходов, с помощью которых были получены символы из алфавита.
Подпроцесс	**Subprocess**	Регулярное выражение, которое является частью регулярного выражения, задающего автомат, и может быть заменено одним символом.
Длина подпроцесса	**Length of the subprocess**	Длина регулярного выражения, задающего подпроцесс.
Алгоритм минимизации конечного автомата	**Minimization algorithm**	Алгоритм, позволяющий получить автомат, эквивалентный исходному, с наименьшим возможным числом состояний.
Метод построения регулярного выражения из конечного автомата с заменой подпроцессов	**Method of constructing regular expression out of an FSM using subprocess changing**	Метод построения регулярного выражения из конечного автомата, при котором переходы отождествляются с регулярными выражениями, повторяющиеся регулярные выражения именуются подпроцессами и заменяются ключевыми символами.
Префикс	**Prefix**	Подстрока, с которой начинается строка символов.

ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

Лист регистрации изменений									
Номера листов (страниц)					Всего листов (страниц в докум.)	№ документа	Входящий № сопроводительного докум. и дата	Подп.	Дата
Изм.	Измененных	Заменен ных	Новых	Аннулированных					

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.502610-01 34				
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата