# Тема проекта

Здравствуйте уважаемая комиссия, я Пронин А.С. из группы ИУ7-82б, и тема моей работы “Метод анализа активности пользователей системы автоматизированного проектирования (САПР) с использованием поиска последовательных шаблонов”.

# Актуальность

Анализ активности пользователей позволяет разработчикам понять, как они взаимодействуют с системой и внести улучшения в интерфейс или функциональность, для повышения эффективности использования системы.

Кроме этого, на основе полученной информации можно сделать выводы на каких аспектах системы стоит сосредоточить усилия.

А также, выявить аномальное или нежелательное поведение, которое может указывать на возможные проблемы взаимодействия пользователя с интерфейсом или ошибки в системе.

# Цель и задачи

На следующем слайде представлены цель и задачи.

# Сравнение рассмотренных методов

В ходе работы мною были рассмотрены различные методы анализа активности пользователей, сравнение которых представлено в таблице на слайде.

Поскольку активность пользователей представляет собой последовательность действий и их характеристик, производящихся в определенный момент времени, то за основу был выбран алгоритм GSP, т.к. он учитывает время совершения транзакций.

# Формализованная постановка задачи

Задача анализа активности пользователей САПР представлена в нотации IDEF0 на слайде. На вход программе подаются информация о выполненных командах и пользовательские параметры: минимальный уровень поддержки, минимальный и максимальный разрывы между командами в секундах. Ограничения на входные данные представлены на экране

Используя методы поиска последовательных шаблонов, система определяет часто встречающиеся последовательности команд, их уровень поддержки и коэффициент зависимости.

# Уровень поддержки

Значение поддержки последовательности равно проценту сессий, которые ее поддерживают.

Сессия поддерживает последовательность, если содержит все ее элементы в том же порядке и временные промежутки между каждыми двумя соседними командами удовлетворяют параметрам, заданным пользователем.

На слайде представлен пример сессии и последовательностей, которые она поддерживает или нет, где min\_gap и max\_gap это минимальный и максимальный разрыв, задаваемый пользователем.

# Коэффициент зависимости

Коэффициент зависимости показывает, насколько команды в последовательности зависят друг от друга и считается как отношение поддержки последовательности к произведению поддержек всех подпоследовательностей, состоящих из 1 команды. Если значение коэффициента <= 1, значит зависимости нету. Если же > 1, то зависимость есть. Чем больше единицы, тем вероятней то, что эти команды использовались вместе.

На слайде представлен пример расчета коэффициента зависимости.

# Ключевые этапы алгоритма

Разработанный метод состоит из двух основных этапов: Генерация кандидатов и Подсчет поддержки кандидатов.

На следующих слайдах представлены схемы для данных этапов.

# Генерация кандидатов

При генерации кандидатов просто перебираются все сочетания текущих последовательностей, и если условие их объединения выполняется, то добавляется новый кандидат.

# Проверка поддержки кандидата сессией

При подсчете поддержки кандидатов каждая сессия проверяется на содержание рассматриваемой последовательности. На данном этапе алгоритм переключается между двумя фазами: поиск следующей и предыдущей команды (по англ. forward и backward phase). Происходит это пока последовательность не будет полностью найдена или же какая-либо команда из нее не будет отсутствовать.

# Структура программного обеспечения

На данном слайде представлена структура ПО.

Модуль преобразования логов записывает данные в таблицу с помощью модуля взаимодействия с базами данных. А модуль вычисления часто встречающихся последовательностей, реализующий разработанный метод, на основе этой базы данных вычисляет результат, для передачи его пользователю через интерфейс.

# Сравнительный анализ времени выполнения метода в зависимости от параметров

В качестве исследования мною был проведён сравнительный анализ времени выполнения разработанного метода в зависимости от его параметров.

Как и ожидалось, при уменьшении минимального уровня поддержки, время выполнения будет расти т.к. в таком случае больше последовательностей будут проходить отбор, что видно по 1ому графику.

На 2ом графике мы видим, что при увеличении минимального разрыва между командами, время выполнения уменьшается т.к. в таком случае получается меньше последовательностей из-за увеличения ограничения.

На последнем графике видно, что при уменьшении максимального разрыва между командами, время выполнения тоже уменьшается, потому что в этой ситуации, также получается меньше последовательностей проходят ограничения.

# Сравнительный анализ времени выполнения этапов метода

Также мною был проведён сравнительный анализ времени выполнения этапов метода, в результате которого, можно сделать вывод, что подсчет поддержки кандидатов занимает большую часть времени, нежели их генерация.

# Заключение

По итогу проделанной работы была достигнута цель и решены все поставленные задачи, перечисленные на экране.

# Перспективы дальнейшего развития

В качестве возможных путей развития данной разработки можно выделить следующие:

* Возможность предсказания следующей команды пользователю, что улучшит эффективность взаимодействия с системой
* Подсчет процента содержания найденных последовательностей в сессиях или
* Вычисление времени, необходимого для выполнения последовательности для повышения информативности результата

# Демонстрация

Спасибо за внимание, теперь можно приступить к вопросам и демонстрации ПО.