РЕФЕРАТ

Курсовая записка содержит страниц, части, 1 приложение, рисунков, таблиц и 4 источника.

Целью курсовой работы являлось создание web ориентированного приложения для управления товарами компьютерного магазина. Поставленная задача решалась используя язык java, базу данных postgresql и технологии JavaEE. В результате поставленная задача была решена, в следствии чего был создан программный продукт подходящий для малых и средних магазинов.

ТОВАР, ПРОИЗВОДИТЕЛЬ, КАТЕГОРИЯ, ХАРАКТЕРИСТИКА, СУБД, ОТЧЁТ, СТАТИСТИКА, ФОРМА.

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ

Задачей нахождения кратчайшего пути является поиск оптимального и короткого пути между двумя точками. Проблема нахождения кратчайших путей возникает в таких случаях как: оптимизация перевозки грузов и пассажиров, оптимальная маршрутиризация пакетов в сети, навигация искусственного интеллекта и игрока в компьютерных играх, а так же навигация роботов в пространстве. Все компьютерные игры имеют поиск путей в том или ином виде, поэтому скорость и точность алгоритма часто влияет на качество искусственного интеллекта и восприятия игрока.

Существует несколько представлений пространства для проведения поиска путей, одним из них является квадратная сетка, которая проста для создания и используется в стратегиях реального времени и других играх с двумерной картой. Хотя квадратная сетка в большинстве случаев является неоптимальным представлением области поиск, с ней очень просто работать и легко модифицировать, что значительно упрощает программную работу с игровой картой. В следствии неоптимальности представления карты появляется необходимость в оптимизации алгоритмов поиска работающих с ней посредством общей оптимизации логики работы алгоритма, введение некоторых допущений и ограничений на область поиска, а так же проведение низкоуровневых оптимизаций.

Таким образом, задача нахождения кратчайшего пути на области представленной квадратной сеткой является актуальной проблемой, которая будет рассмотрена и исследована в данной работе.

Целью выпускной работы является проведение анализа существующих алгоритмов поиска путей, разработка различных вариантов алгоритмов и их оптимизаций, создание оптимизированной универсальной библиотеки для нахождения оптимальных маршрутов и анализ полученных результатов.

0.1 АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

0.1.1 Алгоритмы нахождения кратчайших путей

0.1.1.1 A*

Алгоритм A* (A звёздочка) - это алгоритм общего назначения, который может быть использован для решения многих задач, например для нахождения путей. A* является вариацией алгоритма Дейкстры и используя эвристическую функция для ускорения работы, при этом гарантируя наиболее эффектное использование онной. Алгоритм A* поочерёдно рассматривает наиболее перспективные неисследованные участки. Когда участок исследован, алгоритм останавливается если это конечная точка, иначе все его соседи добавляются в список для дальнейшего исследования.

В алгоритме A* существуют такие понятия: $g(n)$ - точная стоимость пути от начальной точки до точки n, $h(n)$ - эвристическая функция оценивающая расстояние от точки n до конечной точки.

Свойства алгоритма A*:

- Алгоритм гарантирует нахождение пути между точками, если он существует.
- Если эвристическая функция $h(n)$ не переоценивает действительную минимальную стоимость пути, то алгоритм работает наиболее оптимально.
- A* также оптимально эффективен для заданной эвристики $h(n)$.

0.1.1.2 JPS

0.1.1.3 HPA* и HAA*

HPA* (Hierarchical Path-Finding A*) - добавляет алгоритму A* иерархическую абстракцию, разбивая карту на прилегающие друг к другу кластеры, которые соединены входами. Одной из основных идей алгоритма является то, что расчёт пути в A* каждый раз происходит с нуля, что можно исправить добавив сохранение кратчайших путей между определёнными точками.

Первым этапом алгоритма является препроцессинг карты для построения кластеров и их входов. При этом возможно построение нескольких уровней

графа кластеров используя один и тот же алгоритм рекурсивно на созданных во время предыдущего прохода кластерах.

Во время исполнения программы запрос нахождения пути выполняется рекурсивно находя и уточняя путь на графе начиная с самых крупного уровня кластеров. После нахождения пути может применятся его сглаживание.

Алгоритм HPA* работает с такими допущениями:

- Все актёры имеют одинаковый размер, при этом все части навигационной сетки проходимы ими;
- На всех участках карты агенты имеют одинаковую проходимость.

Иерархическая структура карты сильно ускоряет поиск пути, однако алгоритм HPA* работает не учитывая такие важные параметры как размер агентов и проходимость местности.

В итоге размер агентов и проходимость карты должны учитываться при нахождении пути, что в случае с алгоритмом HPA* приводит к тому, что эти параметры должны учитываться при оценки путей между входами.

Алгоритм иерархического поиска является достаточно абстрактным для учёта указанных проблем. Для этого на основе алгоритма HPA* был создан алгоритм HAA* (Hierarchical Annotated A*), который при создании путей между входами кластера учитывает размеры атёров и проходимость местности.

Основная разница с алгоритмом HPA* у алгоритма HAA* состоит в шаге формирования пути между транзитными точками в рамках кластера и дальнейшем шаге их оптимизации. При нахождении пути между транзитными точками в кластере следует найти пути для всех агентов разных размеров и проходимости

В итоге алгоритм HAA* имеет такие же преимущества как HPA*, а так же возможность учёта размера агента и проходимости карты. В зависимости от выбранной тактики устранения похожих путей между транзитными точками кластеров конечный путь будет хуже оптимального до 4-8%.

ВЫВОДЫ

В ходе курсовой работы была разработана web ориентированная система управления компьютерным магазином. Данная система подходит для управления магазинами с низкой и средней загруженностью.

В ходе курсовой работы была разработана архитектура приложения, а так же архитектура базы данных. Были выбраны и изучены такие технологии: jsf, springioc, primefaces, hibernate. В качестве базы данных была выбрана и изучена postgresql.

В результате была создана система, которая позволяет отображать и удобно редактировать товары и их характеристики, производителей, категории. Система реализует удобный и гибкий поиск по товарам. Также система имеет возможность формирования и экспорта отчётов, а также просмотра статистики.

В ходе работы была проанализированна выбранная модель базы данных. В результате был сделан вывод, что данная модель не является оптимальной по производительности и более производительные системы должны быть основаны на nosql решениях или менее гибких sql решениях с таблицами для всех категорий товаров.

СПИСОК ЛИТЕРАТУРЫ

1. John L. Viescas. SQL Queries for Mere Mortals. [Текст] / John L. Viescas, Michael J. Hernandez, 2007. - 672с.

2. David Geary. Core JavaServer Faces (3rd Edition). [Текст] / David Geary, Cay S. Horstmann, 2010. - 672с.

3. James Elliott. Harnessing Hibernate. [Текст] / James Elliott, Timothy M. O'Brien, Ryan Fowler, 2008. - 382с.

4. Oleg Varaksin. PrimeFaces Cookbook. [Текст] / Oleg Varaksin, Mert Caliskan, 2013. - 328с.