**Пояснительная записка**

**1.Введение**

Формирование обособленной и значимой части математического программирования - дискретного программирования – началось в 60-х годах 20 века. Наиболее важная задача данного направления – это выбор оптимального варианта из конечного и, зачастую, очень большого множества. При этом, в случае возникновения задач с предельным количеством значений и вариантов, появляется ряд особенностей, которые почти невозможно встретить в линейных, во многокритериальных или в выпуклых типовых задачах математического программирования. К таким особенностям можно отнести: нерегулярность, трудности при определении окрестности, невозможность выполнения большого перебора на ЭВМ и т. д.

Понятие «дискретная оптимизация» или «дискретное программирование» зачастую употребляется для описания класса оптимизационных задач, которые решаются именно посредством перебора конечного множества вариантов.

Приведем список наиболее распространенных классов задач, дающих хорошее представление о дискретной оптимизации:

1. Комбинаторная оптимизация
2. Целочисленное линейное программирование
3. Комбинаторная геометрия
4. Задачи для потоков в сетях
5. Траекторные задачи
6. Задачи упаковки

Однако, данный перечень было бы ошибочно считать полным и достаточно структурированным, ведь он лишь отражает общепринятые обычаи в использовании типовых терминов в математике, не имеющих жестких стандартов.

Из данного списка можно отдельно выделить комбинаторные методы, так как именно они получили значительное развитие в связи с тем, что на практике существует огромная необходимость решать именно комбинаторные задачи. К таким задачам относятся, например:

1. Задача о рюкзаке
2. Задача коммивояжера
3. Задача о назначениях
4. Задача о раскраске графа

В качестве курсовой работы было решено реализовать программную систему для решения именно этих задач дискретной оптимизации. Моей частью командного проекта является разработка системы для решения и визуализации задачи о ранце.

**2.Назначение и область применения**

Классическая задача о ранце является одной из наиболее популярных и востребованных задач дискретного программирования, что вызвано обширным количеством возможных применений в реальной жизни. Данная задача, включая ее модификации, используется в областях экономики, логистики, криптографии, генетики и прочих сферах.

Наша программная система создана в большей степени для визуализации решений задач дискретной оптимизации, поэтому основным ее назначением является демонстрация пользователям самого выигрышного алгоритма для решения конкретной задачи. В случае части программы, которую выполняла я – это наглядное представление решения задачи о рюкзаке с помощью алгоритма динамического программирования.

Программа может использоваться студентами для ознакомления с задачами и алгоритмами дискретной оптимизации; для проверки и сопоставления их решения данной задачи во время выполнения учебных проектов; а также, чтобы показать детям школьного возраста или людям, не знакомым с алгоритмом, вариант решения задачи о рюкзаке.

**3.Технические характеристики**

**3.1 Постановка задачи и применяемые математические методы**

Пусть пользователь задает или генерирует набор предметов, каждый из которых, помимо названия, определяется двумя параметрами – весом и ценностью. Имеется ранец c некоторой величиной вместимости, которая также задается пользователем. Задача состоит в том, чтобы уложить вещи в рюкзак так, чтобы их ценность была максимальной, причем не превышая емкость (весовое ограничение) ранца.

Математически постановка задачи формулируется следующим образом: пусть имеется n предметов. Для каждого i-го предмета задан его вес 0 > wi и стоимость (ценность) 0 > vi , i = 1, 2, 3, … , n. Где n – число предметов. Задано ограничение на максимальный вес рюкзака ‒ C. Каждый xi может принимать только одно из двух значений: xi =1, если i-й предмет упаковывают в рюкзак, или xi = 0, в противном случае. Требуется выбрать из заданного множества предметов набор с максимальной суммарной стоимостью при одновременном соблюдении ограничения на суммарный вес найденного набора.

На данный момент существует несколько методов, позволяющих находить решение задачи о ранце:

1. Полный перебор

2. Метод ветвей и границ

3. Жадный алгоритм

4. Генетический алгоритм.

5. Метод динамического программирования

Данные методы можно разделить на две крупные категории: точные и приближенные. Скорость работы приближенных алгоритмов очень высока, но они не гарантируют оптимальности решения. Так как у нашей системы образовательная область применения, при которой нет работы с очень большими объемами данных, было принято решение использовать метод динамического программирования – он дает точное решение и является более быстрым, чем прямой перебор.

**3.2 Описание алгоритма**