

Что должна позволять программа.

1. Задавать параметры распределения, метод моделирования, объем выборки.
2. Моделировать выборку (два метода).
3. Изображать гистограмму на фоне теоретического распределения. Выдавать значение χ^2 -square, число степеней свободы (d.f. - degree of freedom) и пороговый уровень значимости (p-level).
4. Получать выборку p-levels. Для этого задавать параметры распределения, которое моделируется, параметры распределения, на согласие с которым проверяется гипотеза, и число повторов. Если это одни и те же параметры распределения, то выдавать, какой истинный уровень значимости соответствует заданному уровню значимости. Если это разные распределения, то выдавать, какая мощность у критерия с заданным уровнем значимости против данной альтернативы.
5. Изображать эмпирическую функцию распределения выборки p-levels на фоне теоретической ф.р. (т.е., функции распределения равномерного на $[0,1]$ распределения).

В дополнении к приведенному выше общему заданию, каждому будет дано дополнительное индивидуальное задание.

Для получения зачета необходимо иметь:

- 1) программу (объектно-ориентированную, с классами)
- 2) описание
- 3) объяснение, каким образом с помощью моделирования можно оценивать вероятность ошибки первого рода для критерия и его мощность.

Промежуточные этапы (указаны верхние границы, можно сдавать и раньше)

К **10 марта** нужно сдать всю математику (протестированное моделирование, χ^2 -square, получение выборки из p-levels, вычисление значимостей и мощностей). Моделирование должно быть протестировано и проверено на диапазоны допустимых значений параметров распределения. Функция распределения p-levels выдается в точках от 0 до 1 с шагом 0.05. Нужно уметь объяснять результаты.

К **14 апреля** должны быть освоены графика в Visual C и техника document-interface, а также реализовано изображение гистограммы на фоне теоретического распределения для промоделированной выборки с использованием полиморфизма при реализации классов для моделирования.

При невыполнении сроков будет дан штраф (дополнительное задание).

Требования к описанию программы

Описание программы состоит из следующих частей:

1. Описание для пользователя.

Все вы когда-либо выступали в роли пользователя. Постарайтесь взглянуть на программу со стороны и ответить на те вопросы, которые могут возникнуть у «простого человека»: с чего начать, как устроено задание параметров, какие между ними зависимости. (не делаем – интерфейс должен быть интуитивно понятным).

2. Описание для программиста.

Подготовка документации для созданной системы классов (достаточно описать только ваши математические классы).

В любом случае, надо начинать с того, что заголовочные файлы должны быть снабжены комментариями.

Если пользоваться системой автоматической генерации документации (стандартно -

<http://en.wikipedia.org/wiki/Doxugen>), то для этих комментариев есть стандарт. Комментировать надо сам класс (его предназначение) и его содержание. Функции типа Get и Set можно не комментировать, если из их названия понятно, с каким полем они связаны.

Вообще, документация должна состоять из след. частей:

Общее описание - зачем это все сделано, для чего предназначено и пр.

Структура (иерархия) классов.

Описание каждого класса - например, просто копии заголовочных файлов, которые уже задокументированы.

Пример использования (в функции main с комментариями, должно компилироваться).

Если делать вручную, то можно просто сформировать WORD-файл, скопировав туда заголовочные файлы для классов и добавив общую часть и примеры. Документацию надо делать только для своих классов - распределение, выборка, хи-квадрат и пр.

3. Математическое описание алгоритмов, используемых в программе.//Устно

4. Описание тестов и их результатов. Требование к тестам: в описании должна содержаться вся информация, с помощью которой эти тесты могут быть повторены. //Для зачета не надо

Комментарии к программе для моделирования распределений

- В математической части программы сделать ошибку можно, в частности, при моделировании, при построении гистограммы, при применении критерия хи-квадрат, при вычислении выборки из p -levels и построении по ней эмпирической функции распределения. Если в результате получается распределение p -levels, не имеющее систематического отклонения от равномерного, то, скорее всего, это означает, что все этапы были реализованы правильно. (Объем выборки из p -levels должен быть существенно больше, чем объем выборки.) В принципе, можно на это надеяться, но практика показывает, что так бывает нечасто. Поэтому лучше проводить предварительное тестирование поэтапно.

- Тестировать функции моделирования можно, например, так: вывести несколько первых реализаций датчика случайных чисел и несколько первых реализаций для моделируемого распределения. Затем, следуя алгоритму (а не тексту функции), сосчитать, какой должна быть реализация.

- Для тестирования правильности применения хи-квадрата, имеет смысл вывести теоретические и эмпирические частоты до объединения интервалов, затем после объединения интервалов, а также число степеней свободы, статистику критерия и p -level. Сумма теоретических и эмпирических частот всегда должна быть равна объему выборки.

- Дальнейшей проверкой правильности будет близость гистограмм моделируемого распределения и выборки, а также не очень большие и не очень маленькие значения критерия хи-квадрат (если вы уже будете уверены, что в моделировании у вас ошибок нет, то эти же самые характеристики будут служить тестом для правильности построения гистограммы и вычисления критерия). Математическое ожидание для случайной величины с распределением хи-квадрат равно числу степеней свободы, т.е. реализации должны колебаться вокруг числа степеней свободы. Естественно, самой сильной проверкой будет равномерное распределение p -levels.

- Гистограмма: гистограмма должна занимать по возможности всю отведенную под нее область как по ширине, так и по высоте. Если у распределения бесконечное число состояний, то при изображении "хвост" распределения можно собирать в один столбик, выделив его цветом. Должны быть указаны минимум и максимум по обеим осям. Функции распределения лучше рисовать линиями – так лучше видно, насколько они совпадают/отличаются.

- Хи-квадрат: Для вычисления критерия хи-квадрат нужно объединять состояния следующим образом. Двигаясь слева направо, объединять состояния так, чтобы для каждого объединенного состояния (включая объединенный "хвост") теоретическая частота была больше некоторой константы (например, $N_{MAX}=5$). (!) Объединение интервалов носит технический характер и используется внутри функции вычисления хи-квадрата. На вид гистограммы как оценки плотности это техническое объединение влиять не должно.

- В программе не должно быть утечек памяти. В режиме debug об утечках памяти свидетельствует появляющаяся в окне output строчка Detecting memory leaks! Если четко соблюдать правила, то утечек не будет. К такого рода правилам относится отведение памяти в конструкторе, освобождение в деструкторе; освобождение памяти перед тем, как написать new, если это new не в конструкторе; инициализация нулем всех неинициализированных указателей (в конструкторе); использование виртуальных деструкторов.

- Проверку гипотез удобно реализовать так: у вас в программе существует понятие основной (нулевой) и альтернативных гипотез (в данном случае нулевая и альтернативная гипотезы отличаются разными значениями параметров заданного распределения). Можно всегда моделировать распределение согласно альтернативной гипотезе и проверять на соответствие нулевой гипотезе (т.е. теоретические частоты всегда вычисляются на основе параметров из нулевой гипотезы). Если вы формально делаете альтернативную гипотезу равной нулевой, то распределение p -levels должно быть равномерным и на его основе можно смотреть, насколько истинная вероятность ошибки I рода отличается от заданного уровня значимости. Если альтернативная гипотеза отличается от нулевой, то распределение P -levels показывает, чему равна мощность критерия против заданной альтернативы при заданных уровнях значимости.