**Лабораторная работа 1**

Использование модульного тестирования при разработке программного обеспечения

Задание: Разработать класс соответственно варианту (выбирается после согласования с преподавателем). Создать не менее 10 unit-тестов для тестирования методов разработанного типа. Код должен содержать поясняющие комментарии.

|  |  |
| --- | --- |
| 1. | Разработать тип для работы с матрицами.  Реализовать методы, позволяющие выполнять операции сложения, вычитания и умножения матриц, предусмотрев возможность их выполнения, в противном случае должно генерироваться исключение. |
| 2. | Разработать тип для работы с матрицами.  Реализовать методы, позволяющие выполнять операции умножения матриц, умножения матрицы на число, предусмотрев возможность их выполнения, в противном случае должно генерироваться исключение. |
| 3. | Разработать тип для работы с матрицами.  Реализовать методы, позволяющие выполнять LU-факторизацию матрицы, предусмотрев возможность их выполнения, в противном случае должно генерироваться исключение. |
| 4. | Разработать тип для работы с матрицами.  Реализовать методы, позволяющие выполнять LLT-факторизацию матрицы, предусмотрев возможность их выполнения, в противном случае должно генерироваться исключение. |
| 5. | Разработать тип для работы с матрицами.  Реализовать методы, позволяющие выполнять LDLT-факторизацию матрицы, предусмотрев возможность их выполнения, в противном случае должно генерироваться исключение. |
| 6. | Разработать тип для работы с матрицами.  Реализовать методы, позволяющие находить определитель матрицы, предусмотрев возможность их выполнения, в противном случае должно генерироваться исключение. |
| 7. | Разработать тип комплексное число.  Реализовать методы, позволяющие находить сумму и произведение чисел, предусмотрев возможность их выполнения, в противном случае должно генерироваться исключение. |
| 8. | Разработать тип вектор.  Реализовать методы, позволяющие расстояние между двумя векторами, в противном случае должно генерироваться исключение. |
| 9. | Разработать тип вектор.  Реализовать методы, позволяющие длину вектора, в противном случае должно генерироваться исключение. |
| 10. | Разработать тип определенный-интеграл.  Реализовать методы, позволяющие вычислять интеграл методом прямоугольников, в противном случае должно генерироваться исключение. |
| 11. | Разработать тип определенный-интеграл.  Реализовать методы, позволяющие вычислять интеграл методом трапеции, в противном случае должно генерироваться исключение. |
| 12. | Разработать тип определенный-интеграл.  Реализовать методы, позволяющие вычислять интеграл методом Моне-Карло, в противном случае должно генерироваться исключение. |
| 13. | Разработать тип метрика качества.  Реализовать метод принимающий два массива, содержащих нули и единицы, и вычисляющий количество Верно положительный (True Positive) и Ложно положительных (False Positive) оценок.  Один вектор, фактические классы некоторого объекта, второй вектор, классы предсказанные некоторым алгоритмом. В случае некорректных данных должно генерироваться исключение. |
| 14. | Разработать тип метрика качества.  Реализовать метод, принимающий два массива, содержащих нули и единицы, и вычисляющий количество Ложно положительный (False Positive) и Ложно отрицательных (False Negative) оценок. Один вектор, фактические классы некоторого объекта, второй вектор, классы предсказанные некоторым алгоритмом. В случае некорректных данных должно генерироваться исключение. |
| 15. | Разработать тип метрика качества.  Реализовать метод, принимающий два массива, содержащих нули и единицы, и вычисляющий количество Верно положительный (True Positive) и Ложно отрицательных (False Negative) оценок.  Один вектор, фактические классы некоторого объекта, второй вектор, классы предсказанные некоторым алгоритмом. В случае некорректных данных должно генерироваться исключение. |