

Дисциплина «Программирование»

Практическое задание №2

29 сентября – 6 октября

«Калькулятор матриц»

Процесс выполнения этого задания состоит из трёх частей:

- 1) реализация программы, согласно описанным в условии требованиям;
- 2) оценивание работ других студентов;
- 3) период «споров».

Период реализации программы:

После выдачи задания Вам необходимо выполнить его и загрузить архив (*.zip) с решением задачи (полностью заархивировать решение, созданное средой разработки) до крайнего срока. В работе строго запрещается указывать ФИО, а также любую другую информацию, которая может выдать авторство работы. В случае выявления факта деанонимизации работы, работа может быть аннулирована.

Период взаимного оценивания:

После окончания срока, отведенного на реализацию программы, начинается период взаимного оценивания. Вам будет необходимо проверить пять работ других студентов, также выполнявших данное задание, согласно критериям оценивания. Проверка осуществляется анонимно: Вы не знаете, чью работу Вы проверяете, также, как и человек, кому принадлежит решение, не знает, кем была проверена его работа. Помимо оценки Вам необходимо указать комментарий к каждому из критериев. В случае, если Вы снижаете балл, необходимо подробно описать, за что именно была снижена оценка. Также рекомендовано писать субъективные комментарии, связанные с тонкостями программной реализации, предлагать автору работу более оптимальные на ваш взгляд решения. Снимать баллы за субъективные особенности реализации запрещено.

Период споров:

По окончании периода взаимного оценивания, в случае если Вы не согласны с оценкой, выставленной одним из проверяющих, Вы можете вступить с этим студентом в анонимный диалог с целью уточнения причин выставления оценки по тому или иному критерию. В случае, если проверяющий не ответил Вам, или вы не пришли к обоюдному решению об изменении оценки, Вы можете поставить флаг, и работа будет рассмотрена одним из преподавателей.

Флаги, поставленные без предварительного обсуждения с проверяющим или после окончания периода выставления флагов, будут отклонены.

Также допустима перепроверка Вашей работы преподавателем. В таком случае оценки других проверявших работу не учитываются.

Дедлайн загрузки работы: 6 октября 23:59

Дедлайн проверки: 9 октября 23:59

Дедлайн обсуждения оценок: 11 октября 23:59

Дедлайн выставления флагов: 12 октября 23:59

Возможность поздней сдачи работы в этом задании предоставляться не будет.

Оценивание:

$O_{\text{итог}} = 0,8 * O_{\text{задание}} + 0,2 * O_{\text{проверки}}$, где $O_{\text{задание}}$ – неокруглённая десятибалльная оценка за решение задания выставленная проверяющими с учётом возможной перепроверки преподавателем, а $O_{\text{проверки}}$ неокруглённая десятибалльная оценка, выставленная студентами, чьи работы вы проверяли. Также в случае, если преподаватель обнаружит, что Вы проверили работы некачественно к Вам могут быть применены санкции в виде штрафа до 3 десятичных баллов от $O_{\text{итог}}$ (в таком случае оценка вычисляется по формуле $O_{\text{итог}} = O_{\text{задание}} - \text{Штраф}$).

Необходимо разработать консольное приложение – «Калькулятор матриц».

Алгоритм работы приложения:

В начале работы программы пользователь выбирает операцию, которую хочет выполнить. Среди доступных операций должны присутствовать:

1. нахождение следа матрицы;
2. транспонирование матрицы;
3. сумма двух матриц;
4. разность двух матриц;
5. произведение двух матриц;
6. умножение матрицы на число;
7. нахождение определителя матрицы.

После выбора операции программа получает необходимые входные данные (размерность матриц, их содержимое), которые могут генерироваться случайным образом (параметры для генерации задаются пользователем) или поступать из консоли. Должны быть реализованы все два способа получения входных данных.

Дополнительные требования (критерии оценивания):

1. Для проверки в систему PeerGrade должен быть загружен архив с решением. Тип проекта должен быть Console Application .NET 5.0. Ожидается, что проверка работы будет проводиться, в среде разработки Microsoft Visual Studio 2019, поэтому в случае выполнения задания с использованием другой среды разработки настоятельно рекомендуется проверить возможность открытия и запуска проекта в Microsoft Visual Studio 2019.
2. Текст программы должен быть отформатирован согласно кодстайлу языка C#¹. Для автоматического форматирования в среде Visual Studio достаточно нажать Ctrl+K, D. Идентификаторы должны соответствовать соглашению о именовании C#². Основным требованием, которое требуется соблюдать в данном задании – это написание имён локальных переменных с использованием camelCasing, а при именовании методов и типов PascalCasing.
3. Программа не должна завершаться аварийно (или уходить в бесконечный цикл) при любых входных данных. При некорректных входных данных программа должна выводить сообщение об ошибке и запрашивать ввод заново.
4. Программа должна быть декомпозирована. Каждый из логических блоков должен быть выделен в отдельный метод. Не строго, но желательно, чтобы каждый метод по длине не превышал 40 строк.
5. Интерфейс программы должен быть понятен. Пользователю должны выводиться подсказки о возможных дальнейших действиях и иные необходимые сообщения. Предполагается, что для успешного использования программы не требуется обращения к исходному коду программы.
6. Должно быть реализовано повторение решения. То есть, после окончания игры, пользователю предлагается завершить программу или начать игру снова.
7. Текст программы должен быть документирован. Названия переменных и методов должны быть на английском языке и отражать суть хранимых значений / выполняемых действий.

¹ <https://docs.microsoft.com/en-us/dotnet/csharp/programming-guide/inside-a-program/coding-conventions>

² <https://docs.microsoft.com/en-us/dotnet/standard/design-guidelines/naming-guidelines>

8. Работа должна быть выполнена в соответствии с заданием, критерии оценивания (“Rubric”) доступны для пред-просмотра в системе “PeerGrade”.
9. Для получения отличной оценки более 8 студенту необходимо реализовать больший функционал, чем описано в задании выше. В данном задании Вам необходимо реализовать операцию для решения СЛАУ методом Гаусса или Крамера. Также дополнительным функционалом считается ввод матриц из текстового файла(формат файла должен быть описан разработчиком программы).
10. Также оценивается общее впечатление, которое производит работа (как с точки зрения пользовательского интерфейса, так и с точки зрения написания кода программы). Эта часть оценки остаётся на усмотрение проверяющего.