**Задание 16.**

Разработать приложение с инкапсулированным классом.

Индивидуальные задачи на инкапсуляцию необходимо решать в режиме разработки «*Windows Forms Application*».

Решение задач должно сводиться к максимально инкапсулированной реализации указанного класса.

При решении задач инкапсуляции необходимо сосредоточиться на составлении свойств и методов для классов и для экземпляров классов, которые предоставляют защищённый доступ к полям классов.

По итогам выполнения работы сдаются строго два файла:

– отчёт, выполненный в текстовом редакторе *Microsoft Office Word* (*\*.doc* или *\*.docx*);

– файл с архивом всего решения (*Solution*), написанного на языке *Visual C#* в среде *Microsoft Visual Studio* в режиме *Windows Forms Application* (*\*.zip* или *\*.rar*).

Отправленные поодиночке файлы проверке не подлежат. При отсутствии одного из упомянутых файлов зачёт по заданию не выставляется.

Раздел «Подтверждение соответствия графического пользовательского интерфейса требованиям к оформлению» наполнить демонстрацией скриншотами выполнения каждого из пунктов раздела «Требования к оформлению» из данной постановки задачи.

**Требования к именам файлов:**

**Общий вид формата имени файла:** «*Дата. Задание. Фамилия.mcdx*»

**Формат записи даты:** «*ГГГГММДД*», где *ГГГГ* – четыре цифры текущего года, *ММ* – две цифры текущего месяца, *ДД* – две цифры текущего дня.

**Формат записи задания:** «Задание *NNk*», где *NN* – две цифры номера задания, *k* – обозначение «о», если файл содержит общую часть; обозначение «и», если файл содержит индивидуальную часть; обозначение «ои», если файл содержит как общую, так и индивидуальную части.

**Если устранить замечания по работе удаётся в тот же день:** после фамилии ставится пробел и в круглых скобках записывается номер попытки исправления.

**Примеры правильных имён файлов, которые сдаются на проверку впервые\*:**

«*20240501. Задание 16и. Иванов.docx*»

«*20240501. Задание 16и. Иванов.rar*»

«*20240501. Задание 15и. Иванов.vsdx*»

**Примеры правильных имён файлов, которые сдаются на проверку повторно в тот же день\*:**

«*20240501. Задание 16и. Иванов (1).docx*»

«*20240501. Задание 16и. Иванов (1).rar*»

«*20240501. Задание 16и. Иванов (1).vsdx*»

\* Правила распространяются только при обмене файлами через электронную почту. При работе на платформе *GitHub* устранение замечаний проводится в одних и тех же файлах без необходимости их переименования.

**Внимание!** Не забудьте выполнить автоматическую нумерацию страниц в отчёте.

Отчёт по выполненной работе должен содержать:

0. Титульный лист.

1. Формулировку цели работы.

2. Описание задачи согласно выданному варианту.

3. Составление диаграммы классов, входящих в состав решения.

4. Составление сети Петри запрограммированного технологического процесса.

5. Составление схем алгоритмов методов в составе решения, отмеченных на сети Петри в качестве «эффектов» (метка ).

6. Подбор тестовых примеров.

7. Листинг (код) составленного программного обеспечения.

8. Графический пользовательский интерфейс программного обеспечения и его описание.

9. Подтверждение соответствия графического пользовательского интерфейса требованиям к оформлению.

10. Расчёт тестовых примеров с использованием составленного программного обеспечения.

11. Формулировку вывода о проделанной работе.

Рекомендации к отчёту, доказывающие самостоятельность выполнения работы и упрощающие процедуру проверки отчёта преподавателем:

1. Выполнение дополнительных скриншотов для случаев, когда текстовое описание проделанных действий становится громоздким или трудным к восприятию.

2. Нумерация рисунков (если есть) с подписями, содержащими названия рисунков, например, «Рисунок 1 – Схема алгоритма программы».

**Цель работы (одна из возможных формулировок)**: «Освоение навыков программирования инкапсулированных классов».

**Требования к оформлению**:

1. Заголовок главной формы должен содержать надпись вида: «Задание №15 выполнил: [Фамилия И.О. автора]; Номер варианта: [Номер]; Дата выполнения: [дд/мм/гггг]».

2. Дата выполнения проставляется в момент, когда программа считается законченной и по ней можно готовить итоговый отчёт.

3. Предусмотреть экранный список, в который выводятся все созданные экземпляры класса по формату, предусмотренному переопределённым методом *ToString()*.

4. На экранной форме должны быть предусмотрены кнопки для добавления нового экземпляра класса, удаления выбранного в списке экземпляра класса, редактирования выбранного в списке экземпляра класса.

5. Добавление и редактирование проводить в дочерней экранной форме, где для чётных вариантов разместить один ярлык и одно поле для ввода текста, в котором параметры перечисляются в установленном порядке с разделителем «;». Для нечётных вариантов предусмотреть ярлыки и поля для ввода текста индивидуально под каждый параметр экземпляра класса.

6. Предусмотреть по нажатии клавиши «*F1*» вызов дочерней экранной формы со справкой о работе с программой, текст для которой необходимо считывать из файла в статический по размеру ярлык (*label*).