

### Общие сведения

**Цели учебной практики (технологической (проектно-технологической) практики):** изучение и освоение базовых понятий, методов и приемов использования инструментальных средств и технологий программирования при решении практических задач с выбором различных структур данных и организацией программного графического интерфейса пользователя, закрепление и приобретение новых знаний и практических навыков программирования.

**Задачи учебной практики (технологической (проектно-технологической) практики):**

- формирование базовых практических понятий, лежащих в основе процесса разработки программного графического интерфейса пользователя,
- получение навыков применения средств визуализации при решении практических задач и использовании различных структур данных,
- освоение способов реализации программ на выбранном рабочем языке программирования C++ (C#) с учётом особенностей реализации в конкретной системе программирования.

**Студент в ходе учебной практики (технологической (проектно-технологической) практики):**

- самостоятельно выбирает язык программирования (C++ или C#), разные индивидуальные задания могут быть реализованы на разных языках программирования,
- самостоятельно выбирает систему или системы программирования для разных заданий,
- самостоятельно проектирует графический интерфейс пользователя (структурный состав элементов, их размещение, свойства и т.п.),
- должен изучить предложенный ознакомительный материал и реализовать приложения по нему и для 2 индивидуальных заданий,
- готовит "Отчет по учебной практике (технологической (проектно-технологической) практике)" (шаблон представлен на сайте вуза - традиционно титульный лист, лист задания, аннотация на русском и английском языках) из 3 разделов согласно заданиям (по ознакомительному материалу и 2 индивидуальным заданиям) и итоговому выводу.

**Комментарий:** ознакомительный материал рассчитан на создание приложения Windows Forms в MVS. Однако целью является графический оконный интерфейс и элементарная машинная графика - поэтому возможно использование иных вариантов реализации (не Windows Forms в MVS и вообще не Microsoft), но и в этом случае надо выполнить шаги первого раздела по аналогии с ознакомительным материалом и тогда в разделах отчета дополнительно описывать используемые возможности иного средства разработки.

**Текущий контроль:** промежуточные итоги выполнения индивидуальных заданий и защиты отчетов подводятся в контрольные сроки точек текущего контроля и влияют на итоговую оценку (при отсутствии результатов и раздела отчета на контрольную дату итоговая оценка снижается на 1 балл, при частичных недостатках на контрольную дату идет накопительное снижение балла в зависимости от невыполненных составляющих задания).

Оценка с ранжированием выставляется по всей совокупности результатов по заданиям и оформлению отчета по каждому разделу с учетом сроков выполнения по точкам текущего контроля, максимальная оценка - отлично.

Задания на конец учебной практики (технологической (проектно-технологической) практики) должны быть выполнены ВСЕ (полностью или частично), при отсутствии результатов хотя бы по одному из разделов отчет считается не подготовленным полностью с итоговой неаттестацией.

Учебная практика  
(технологическая (проектно-технологическая) практика)

**График точек текущего контроля в 2023 (можно идти по опережающему графику):**

- ознакомительный материал: 04.07
- 1-е индивидуальное задание: 07.07
- 2-е индивидуальное задание: 11.07
- итоговый отчет: 12.07

По каждому заданию представляется в электронном виде отчетный раздел до указанного срока, чтоб можно было внести коррективы в случае указания на доработку.

Не запрещается подготовить сразу итоговый отчет, но он должен покрывать контрольные сроки:

- на 04.07 - в нем все разделы
- на 04.07 только ознакомительный материал и на 07.07 все оставшиеся разделы
- на 04.07 только ознакомительный материал, на 07.07 добавка по 1-му индивидуальному и на 12.07 весь отчет
- на 04.07 ознакомительный материал и по 1-му индивидуальному, на 07.07 весь отчет
- ...

**Содержательные составляющие учебной практики (технологической (проектно-технологической) практики)**

1. Основные возможности инструментальных средств и технологий программирования при организации программного графического интерфейса пользователя. Понятие оконного интерфейса. Понятие события и способы его обработки.
2. Основные визуальные компоненты при организации оконного интерфейса и их свойства. Библиотеки визуальных компонентов и их использование.
3. Основные диалоговые инструменты и способы их использования. Создание меню и его элементов.
4. Работа с примитивными элементами, обеспечивающими создание оконного интерфейса.
5. Использование графических возможностей при программировании оконного интерфейса.
6. Графическое представление структур данных разного вида и визуализация основных операций с ними. Ограниченный размер окна, масштабирование.

*Выполнение индивидуальных заданий направлено на:*

1. Создание простого окна с примитивными характеристиками и элементами оформления.
2. Программирование общих оконных элементов управления (кнопки разного вида, диалоговые элементы и др.) на основе обработки событий.
3. Программирование элементов управления и диалога.
4. Проектирование и использование меню.
5. Использование основных графических примитивов и задание их свойств для визуализации геометрических объектов.
6. Визуализация структур данных разного вида и операций по изменению представленных ими данных.

**Комментарии по выполнению заданий**

1. Подготовка приложения или отдельных приложений на основе ознакомительного материала

Предлагается выполнить последовательность действий (**файлы work++ или work#**) по использованию элементов оформления пользовательского интерфейса. Каждое действие может быть оформлено отдельным приложением

Учебная практика  
(технологическая (проектно-технологическая) практика)

или все действия сосредоточены в одном приложении. Дополнительно надо реализовать пример действий по формированию движения по траектории (**файл DynGO**) и выявить какие характеристики отвечают за объект на траектории. Наполнение соответствующих элементов может быть произвольным.

*Раздел отчета содержит:* по каждому пункту только результаты (скриншоты) по каждому из предложенных действий, демонстрирующих ситуации до и после выполнения действия (иногда несколько этапов в процессе выполнения как при движении по траектории), в отдельных случаях полезно добавить картинку установки отдельных параметрических настроек.

## 2. Подготовка приложения по 1-му индивидуальному заданию

По ранее реализованному образцу действий по формированию движения по траектории необходимо реализовать приложение, которое обеспечивает движение заданного объекта по заданной траектории.

Элементы управления и настройки, меню и все подходящее под параметрическую настройку должны обеспечить максимальное задание всех мыслимых свойств используемых в задании объектов, в частности:

- выбор цветов (объекта и траектории)
- выбор толщины линий (объекта и траектории)
- выбор вида линий (сплошная, пунктирная, штриховая и т.д.)
- пульсирование размера объекта (от min к max) с разной скоростью
- выбор скорости движения объекта по траектории (например, числовое задание или из выпадающего списка)
- в отдельных заданиях выбор стиля и цвета заливки
- определение скорости движения объекта по траектории
- число повторов прохождения объекта по траектории
- шаг перемещения объекта по траектории
- шаг изменение угла объекта при его вращении вокруг центра (вершины, ключевой точки и т.п.)
- направление движение объекта (по/против часовой стрелки или вперед/назад и т.п.)
- и все остальное приходящее в голову по характеристикам объекта или движения ...

Упрощенный набор характеристик приведет к снижению оценки!

Траектория и объект могут выходить за область прорисовки в зависимости от задаваемых параметров.

Естественно надо выбрать размещение области прорисовки и расположения элементов интерфейса (кнопок, меню и пр.). Для контроля в окно помещается картинка из файла задания.

*Раздел отчета содержит:* формулировку задания, математическую постановку, описание используемых элементов для интерфейса и их настройку (свойства), описание используемых графических примитивов, текст программы (можно меньшим размером шрифта, чем основной текст), примеры работы (скриншоты в динамике движения объекта) с учетом изменения параметров, выводы

## 3. Подготовка приложения по 2-му индивидуальному заданию

Формирование графического изображения на основе математического описания фрактала - правила его построения сразу требуют наличия привычного из курса "Структуры данных" дерева, у которого для каждого узла число потомков согласуется с правилом выполнения очередного шага при построении фрактала.

Учебная практика  
(технологическая (проектно-технологическая) практика)

В вариантах указывается как должен был прорисован фрактал - сразу на конкретном шаге его построения или должна быть представлена последовательность шагов его построения. В итоге каждому шагу построения должен соответствовать уровень дерева.

Должны быть нарисованы как фрактал, так и дерево - естественно надо выбрать размещение области прорисовки и расположения элементов интерфейса (кнопок, меню и пр.), в заданиях указаны требования по расположению фракталов (одиночно конкретного шага построения фрактала или набор фракталов по каждому шагу). Местоположение изображения дерева выбирается самостоятельно. Для контроля в окно помещается картинка из файла задания.

Каждый уровень дерева на изображении должен иметь свой цвет и на фрактале для его элементов, появляющихся на определенном шаге этого уровня дерева должен использоваться тот же цвет. Дерево и фрактал должны быть разноцветными, за исключением ситуации, когда по заданию фрактал должен быть изображен сразу на конкретном шаге построения, но и в этом случае можно подумать как разнообразить цветом.

В итоге, элементов управления и настройки, меню и все подходящее под параметрическую настройку должно быть минимальное число только под обеспечение выполнения задания. В частности, можно задавать масштаб изображения с прорисовкой не на все окно.

**Существенное требование:** в начале в памяти формируется дерево, каждый узел которого содержит рассчитываемые координаты узла дерева ( $x$  и  $y$ ) и возможно какой-то части фрактала (отдельные переменные) по размеру окна или его части, куда надо поместить фрактал и дерево, далее начинается обход дерева с построением соответствующего изображения фрактала и самого дерева. это облегчит задачу масштабирования изображений. Отсутствие предварительного расчета координат с их явным хранением в узлах дерева приведет к снижению оценки!

*Раздел отчета содержит:* формулировку задания, математическую постановку, описание используемых элементов для интерфейса и их настройку (свойства), описание используемых графических примитивов, текст программы (можно меньшим размером шрифта, чем основной текст), примеры работы (скриншоты в динамике прорисовки дерева и фрактала в разных ситуациях, если не применены возможности по "закадровому" формированию изображения с его "мгновенным" выводом) с учетом задания значений немногочисленных параметров, выводы