



Федеральное агентство по рыболовству
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Калининградский государственный технический университет»
(ФГБОУ ВО «КГТУ»)

УТВЕРЖДАЮ
Начальник УРОПС

Фонд оценочных средств
(приложение к рабочей программе модуля)
«ВЫСОКОУРОВНЕВЫЕ ТЕХНОЛОГИИ ПРОГРАММИРОВАНИЯ»

основной профессиональной образовательной программы бакалавриата
по направлению подготовки

09.03.03 ПРИКЛАДНАЯ ИНФОРМАТИКА

Профиль программы
«ПРИКЛАДНАЯ ИНФОРМАТИКА В ЭКОНОМИКЕ»

ИНСТИТУТ
РАЗРАБОТЧИК

цифровых технологий
кафедра прикладной информатики

1 РЕЗУЛЬТАТЫ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Таблица 1 – Планируемые результаты обучения по дисциплине, соотнесенные с установленными индикаторами достижения компетенций

| Код и наименование компетенции | Индикаторы достижения компетенции | Дисциплина | Результаты обучения (владения, умения и знания), соотнесенные с компетенциями/индикаторами достижения компетенции |
|--|---|--|---|
| <p>ОПК-7: Способен разрабатывать алгоритмы и программы, пригодные для практического применения</p> <p>ПКС-2 Выполнение работ и управление работами по созданию (модификации) и сопровождению ИС, автоматизирующих задачи организационного управления и бизнес-процессы</p> <p>ПКС-9 Способен программировать приложения и создавать программные прототипы решения прикладных задач</p> | <p>ОПК-7.3: Владеет навыками программирования, отладки и тестирования прототипов программно-технических комплексов задач</p> <p>ПКС-2.6 Организационное и технологическое обеспечение кодирования на языках программирования</p> <p>ПКС-9.1 Программирует приложения и создает программы, использующие объектно-ориентированную технологию программирования</p> | <p>Высокоуровневые технологии программирования</p> | <p><u>Знать:</u></p> <ul style="list-style-type: none"> - законы эволюции программного обеспечения; - сущность объектно-ориентированного подхода к проектированию и разработке программ в среде событийного и визуального программирования; - базовые понятия визуального и событийного программирования; - структуру современного программного интерфейса с базами данных <p><u>Уметь:</u></p> <ul style="list-style-type: none"> - проводить анализ современных методов и средств программирования в процессе их выбора при решении прикладных задач различных классов; - использовать преимущества высокоуровневых технологий при создании программных приложений; - тестировать и отлаживать программные системы, реализованные на основе современных технологий программирования. <p><u>Владеть:</u></p> <ul style="list-style-type: none"> - навыками разработки программных приложений на основе современных высокоуровневых технологий. |

2 ПЕРЕЧЕНЬ ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПОЭТАПНОГО ФОРМИРОВАНИЯ РЕЗУЛЬТАТОВ ОСВОЕНИЯ ДИСЦИПЛИНЫ (ТЕКУЩИЙ КОНТРОЛЬ) И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

2.1 Для оценки результатов освоения дисциплины используются:

- оценочные средства текущего контроля успеваемости;
- оценочные средства для промежуточной аттестации по дисциплине.

2.2 К оценочным средствам для текущего контроля успеваемости относятся:

- задания и контрольные вопросы по лабораторным работам;
- тестовые задания.

2.3 К оценочным средствам для промежуточной аттестации по дисциплине, проводимой в форме экзамена, относятся:

- задания и контрольные вопросы по курсовой работе;
- задание на контрольную работу для студентов заочной формы обучения
- экзаменационные вопросы.

3 ОЦЕНОЧНЫЕ СРЕДСТВА ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ

3.1 Тестовые задания используются для оценки освоения тем дисциплины студентами очной формы обучения – знания основных понятий и средств информатизации деятельности предприятий и организаций (Приложение № 1). Тестирование обучающихся проводится на занятиях после рассмотрения на лекциях соответствующих тем.

Тестирование предусматривает выбор правильного ответа на поставленный вопрос из четырех предлагаемых вариантов ответа.

Оценка («отлично», «хорошо», «удовлетворительно» или «неудовлетворительно») по итогам тестирования выставляется в соответствии с критериями, указанными в табл. 1.

Таблица 1 – Система и критерии оценивания зачетного тестирования

| Система оценок | 2 | 3 | 4 | 5 |
|---|---|---|---|---|
| | 0-40% | 41-60% | 61-80 % | 81-100 % |
| Критерий | «неудовлетворительно» | «удовлетворительно» | «хорошо» | «отлично» |
| Системность и полнота знаний в отношении изучаемых объектов | Обладает частичными и разрозненными знаниями, которые не может научно-корректно связывать между | Обладает минимальным набором знаний, необходимым для системного взгляда на изучаемый объект | Обладает набором знаний, достаточным для системного взгляда на изучаемый объект | Обладает полнотой знаний и системным взглядом на изучаемый объект |

| | | | | |
|----------------|---|---------------------|----------|-----------|
| Система оценок | 2 | 3 | 4 | 5 |
| | 0-40% | 41-60% | 61-80 % | 81-100 % |
| Критерий | «неудовлетворительно» | «удовлетворительно» | «хорошо» | «отлично» |
| | собой (только некоторые из которых может связывать между собой) | | | |

3.2 В приложении № 2 приведены типовые задания и контрольные вопросы по лабораторным работам, предусмотренным рабочей программой дисциплины. Оценка результатов выполнения задания по каждой лабораторной работе производится при представлении студентом отчета по лабораторной работе и на основании ответов студента на вопросы по тематике лабораторной работы. Студент, самостоятельно выполнивший задание и продемонстрировавший знание использованных им средств и приемов программирования задачи получает по лабораторной работе оценку «зачтено».

4 ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ДИСЦИПЛИНЕ

4.1 Задание по контрольной работе, выполняемой студентами заочной формы обучения, предусматривает создание индивидуального проекта информатизации предприятия (Приложение № 3). Результаты контрольной работы позволяют оценить успешность освоения студентами тем дисциплины.

Оценка контрольной работы определяется количеством допущенных в ней ошибок:

- «отлично» - ошибок нет;
- «хорошо» - не более двух ошибок в планировании проекта;
- «удовлетворительно» - при трех ошибках, допущенных при планировании проекта;
- «неудовлетворительно» - более трех ошибок или студент полностью не справился с заданием.

4.2 Пример задания на курсовую работу представлен в приложении № 4.

По результатам защиты курсовой работы (студент представляет результаты программирования в виде пояснительной записки заданной структуры, демонстрирует работу программы в различных режимах и отвечает на вопросы преподавателей) выставляется экспертная оценка («отлично», «хорошо», «удовлетворительно» или «неудовлетворительно»), которая учитывается при итоговой аттестации по дисциплине (на экзамене).

4.3 Промежуточная аттестация по дисциплине проводится в форме экзамена.

Оценка («отлично», «хорошо», «удовлетворительно» или «неудовлетворительно») выставляется студенту, успешно выполнившему лабораторные работы в течение семестра обучения и получившему соответствующую оценку по результатам тестирования. Студент, не выполнивший лабораторный практикум, получает оценку «неудовлетворительно». Студент, выполнивший лабораторный практикум, но имеющий неудовлетворительную оценку по результатам тестирования в семестре проходит тестирование повторно.

4.4 Промежуточная аттестация по дисциплине проводится в форме зачета и экзамена. Промежуточная аттестация в форме зачета проходит по результатам прохождения всех видов текущего контроля успеваемости. В приложении № 4 приведены вопросы к экзамену по дисциплине.

5 СВЕДЕНИЯ О ФОНДЕ ОЦЕНОЧНЫХ СРЕДСТВ И ЕГО СОГЛАСОВАНИИ

Фонд оценочных средств для аттестации по дисциплине «Высокоуровневые технологии программирования» представляет собой компонент основной профессиональной образовательной программы бакалавриата по направлению подготовки 09.03.03 Прикладная информатика.

Фонд оценочных средств рассмотрен и одобрен на заседании кафедры прикладной информатики (протокол № 4 от 14.10.2022 г.).

И.О. заведующего кафедрой



М.В Соловей

Приложение № 1

ТЕСТОВЫЕ ЗАДАНИЯ ПО ДИСЦИПЛИНЕ

Вариант №1

Вопрос 1. Класс - это:

- любой тип данных, определяемый пользователем;
- тип данных, определяемый пользователем и сочетающий в себе данные и функции их обработки;
- структура, для которой в программе имеются функции работы с ней;
- элемент алгоритма.

Вопрос 2. Членами класса могут быть:

- как переменные, так и функции, могут быть объявлены как `private` и как `public`;
- только переменные, объявленные как `private`;
- только функции, объявленные как `private`;
- только переменные и функции, объявленные как `private`;
- только переменные и функции, объявленные как `public`.

Вопрос 3. Конструктор – это:

- метод, имя которого совпадает с именем класса и который вызывается автоматически при создании объекта класса;
- метод, имя которого совпадает с именем класса и который вызывается автоматически при объявлении класса (до создания объекта класса);
- метод, имя которого необязательно совпадает с именем класса и который вызывается при создании объекта класса;
- метод, имя которого совпадает с именем класса и который необходимо явно вызывать из головной программы при объявлении объекта класса.

Вопрос 4. Объект – это:

- переменная, содержащая указатель на класс;
- экземпляр класса;
- класс, который содержит в себе данные и методы их обработки;
- динамическая переменная.

Вопрос 5. Следующие утверждения являются правильными:

- конструкторы класса не наследуются;
- конструкторов класса может быть несколько, их синтаксис определяется программистом;
- конструкторов класса может быть несколько, но их синтаксис должен подчиняться правилам перегрузки функций;
- конструктор возвращает указатель на объект;
- конструктор не возвращает значение.

Вопрос 6. Деструктор - это

- метод, который уничтожает объект;
- метод, который удаляет объект;
- метод, который освобождает память, занимаемую объектом;

- системная функция, которая освобождает память, занимаемую объектом.

Вопрос 7. Следующие утверждения являются правильными:

- у конструктора могут быть параметры
- конструктор наследуется, но должен быть перегружен
- конструктор должен явно вызываться всегда перед объявлением объекта
- конструктор вызывается автоматически при объявлении объекта
- объявление каждого класса должно содержать свой конструктор
- если конструктор не создан, компилятор создаст его автоматически

Вопрос 8. Следующие утверждения являются правильными:

- деструктор - это метод класса, применяемый для удаления объекта;
- деструктор - это метод класса, применяемый для освобождения памяти, занимаемой объектом;
- деструктор - это отдельная функция головной программы, применяемая для освобождения памяти, занимаемой объектом;
- деструктор не наследуется;
- деструктор наследуется, но должен быть перегружен.

Вопрос 9. Наследование -

- это механизм, посредством которого производный класс получает элементы родительского и может дополнять либо изменять их свойства и методы
- это механизм переопределения методов базового класса
- это механизм, посредством которого производный класс получает все поля базового класса
- это механизм, посредством которого производный класс получает элементы родительского, может их дополнить, но не может переопределить.

Вопрос 10. Правильное объявление производного класса -

- class MoreDetails:: Details;
- class MoreDetails: public class Details;
- class MoreDetails: public Details;
- class MoreDetails: class(Details);

Вопрос 11. Событие в виде нажатия правой клавиши мыши в языке Python обозначается как _____

Вопрос 12. Многострочным редактором является виджет _____

Вопрос 13. Для очистки всей канвы Canvas используется метод _____ с аргументом ALL

Вопрос 14. Свойством _____ задается заголовок у формы

Вопрос 15. Нумерация строк в редакторе Text начинается с ____

Вопрос 16. Программное занесение текста в редактор Text происходит через метод _____

Вопрос 17. В записи `fg = "555"` отсутствует символ _____

Вопрос 18. Главное меню привязывается к форме через свойство _____

Вопрос 19. Объект в ООП является экземпляром _____

Вопрос 20. В редакторе Text включает/отключает перенос строки свойство _____

Вариант № 2

Вопрос 1. Правильными утверждениями являются следующие:

- если элементы класса объявлены как `private`, то они доступны только наследникам класса, но не внешним функциям;
- если элементы класса объявлены как `private`, то они недоступны ни наследникам класса, ни внешним функциям;
- если элементы объявлены как `public`, то они доступны наследникам класса, но не внешним функциям;
- если элементы объявлены как `public`, то они доступны и наследникам класса, и внешним функциям.

Вопрос 2. Возможность и способ обращения производного класса к элементам базового определяется:

- ключами доступа: `private`, `public`, `protected` в теле производного класса
- только ключом доступа `protected` в заголовке объявления производного класса
- ключами доступа: `private`, `public`, `protected` в заголовке объявления производного класса
- ключами доступа: `private`, `public`, `protected` в теле базового класса

Вопрос 3. Какие соответствия являются правильными между спецификатором базового класса, ключом доступа в объявлении производного класса и правами доступа производного класса к элементам базового:

- ключ доступа - `public`; в базовом классе: `private`; права доступа в производном классе - `protected`
- ключ доступа - любой; в базовом классе: `private`; права доступа в производном классе - нет прав
- ключ доступа - `protected` или `public`; в базовом классе: `protected`; права доступа в производном классе - `protected`
- ключ доступа - `private`; в базовом классе: `public`; права доступа в производном классе - `public`
- ключ доступа – любой; в базовом классе: `public`; права доступа в производном классе – такие же, как ключ доступа

Вопрос 4. Дружественная функция - это

- функция другого класса, среди аргументов которой есть элементы данного класса
- функция, объявленная в классе с атрибутом `friend`, но не являющаяся членом класса;
- функция, являющаяся членом класса и объявленная с атрибутом `friend`;
- функция, которая в другом классе объявлена как дружественная данному

Вопрос 5. Правильные утверждения следующие:

- одна функция может быть дружественной нескольким классам
- дружественная функция не может быть обычной функцией, а только методом другого класса
- дружественная функция объявляется внутри класса, к элементам которого ей нужен доступ
- дружественная функция не может быть методом другого класса

Вопрос 6. Шаблон функции - это...

- определение функции, в которой типу обрабатываемых данных присвоено условное обозначение
- прототип функции, в котором вместо имен параметров указан условный тип
- определение функции, в котором указаны возможные варианты типов обрабатываемых параметров
- определение функции, в котором в прототипе указан условный тип, а в определении указаны варианты типов обрабатываемых параметров

Вопрос 7. Правильные утверждения следующие:

- по умолчанию члены класса имеют атрибут `private`
- по умолчанию члены класса имеют атрибут `public`;
- члены класса имеют доступ только к элементам `public`;
- элементы класса с атрибутом `private` доступны только членам класса

Вопрос 8. Переопределение операций имеет вид:

- имя_класса, ключевое слово `operation`, символ операции
- имя_класса, ключевое слово `operator`, символ операции, в круглых скобках могут быть указаны аргументы
- имя_класса, ключевое слово `operator`, список аргументов
- имя_класса, два двоеточия, ключевое слово `operator`, символ операции

Вопрос 9. Для доступа к элементам объекта используются:

- при обращении через имя объекта – точка, при обращении через указатель – операция «->»
- при обращении через имя объекта – два двоеточия, при обращении через указатель – операция «точка»
- при обращении через имя объекта – точка, при обращении через указатель – два двоеточия
- при обращении через имя объекта – два двоеточия, при обращении через указатель – операция «->»

Вопрос 10. Полиморфизм реализован через механизмы:

- перегрузки функций, виртуальных функций, шаблонов
- перегрузки функций, наследования методов, шаблонов;
- наследования методов, виртуальных функций, шаблонов
- перегрузки функций, наследования, виртуальных функций.

Вопрос 11. Метод _____ устанавливает связь виджета проекта с клавиатурой компьютера

- Вопрос 12. Индекс текущего выбора в виджете Combobox определяется через метод _____
- Вопрос 13. Проверяет, все ли символы в строке являются буквами, функция (метод) _____
- Вопрос 14. Разветвления в меню создаются с помощью метода _____
- Вопрос 15. При отсутствии выбора в компоненте Combobox значение свойства current() равно _____
- Вопрос 16. Вид курсора над виджетом задается свойством виджета _____
- Вопрос 17. Модуль tkinter различает _____ (укажите количество) менеджеров геометрии
- Вопрос 18. Для создания горизонтальной разделительной линии в списке опций меню надо использовать метод _____
- Вопрос 19. Кнопка _____ может фиксироваться в нажатом состоянии
- Вопрос 20. Атрибут _____ задает взаимное расположение надписи и рисунка на кнопке

Вариант №3

- Вопрос 1. Полиморфизм – это :
- средство, позволяющее использовать одно имя для обозначения действий, общих для родственных классов
 - средство, позволяющее в одном классе использовать методы с одинаковыми именами;
 - средство, позволяющее в одном классе использовать методы с разными именами для выполнения одинаковых действий
 - средство, позволяющее перегружать функции для работы с разными типами или разным количеством аргументов.
- Вопрос 2. Виртуальными называются функции:
- функции базового класса, которые могут быть переопределены в производном классе
 - функции базового класса, которые не используются в производном классе;
 - функции базового класса, которые не могут быть переопределены в базовом классе;
 - функции производного класса, переопределенные относительно базового класса
- Вопрос 3. Выберите правильный вариант выделения динамической памяти под переменную X типа float:
- float *ptr = new float; X = *ptr;
 - float & ptr = new float; X = & ptr;
 - float * ptr = &X; X = new float;

- X = new float.

Вопрос 4. Полиморфизм в объектно-ориентированном программировании реализуется:

- через механизмы перегрузки (функций и операций), виртуальные функции и шаблоны
- через механизмы перегрузки (функций и операций) и шаблоны;
- через виртуальные функции и шаблоны;
- через механизмы перегрузки (функций и операций) и виртуальные функции

Вопрос 5. В перечислении атрибутом кнопки является:

show
selectcolor
value
side
selectmode

Вопрос 6. Стиль оформления выделенного фрагмента текста в редакторе Text задается в методе _____

Вопрос 7. Изображение загружается в контейнер Canvas через метод _____

Вопрос 8. Метод tag_add() принадлежит классу виджетов _____

Вопрос 9. У виджета Scale() параметр, определяющий интервал, через который отображаются метки шкалы, называется _____

Вопрос 10. Значение active является значением свойства _____

Вопрос 11. ООП – это

Вопрос 12. Метод _____ переводит всю строку текста в верхний регистр

Вопрос 13 Событие в виде нажатия средней клавиши мыши (колесика) в языке Python обозначается как _____

Вопрос 14. Экземпляром класса в ООП является _____

Вопрос 15. Связь виджета проекта с клавиатурой компьютера называется _____

Вопрос 16 Для конвертации строки в целое число служит функция _____

Вопрос 17. При отпускании нажатой кнопки Button она переходит в состояние _____

Вопрос 18. Ввод информации в приложение, при котором информация не отображается на экране, называется _____

Вопрос 19. Синонимом события "Key" является событие _____

Вопрос 20. Обратным для события KeyPress является событие _____

ТИПОВЫЕ ЗАДАНИЯ И КОНТРОЛЬНЫЕ ВОПРОСЫ ПО ЛАБОРАТОРНЫМ РАБОТАМ

Лабораторная работа № 1 Основные элементы пользовательского графического интерфейса

Пример задания

На форме располагаются компоненты: на середине формы метка Label с указанием фамилии студента; Listbox с опциями-цифрами 1, 2, 3, 4; четыре кнопки со стрелками , , , ; редактор Entry; Checkbutton и кнопка «Заккрыть». При нажатии мышью кнопки со стрелкой метка с фамилией смещается в соответствующем направлении на количество пикселей, указанных в Listbox. Текущий шаг перемещения отображается в редакторе Entry, если он виден. Режим его видимости задается флагом Checkbutton.

Контрольные вопросы:

1. Как расшифровывается и переводится GUI?
2. Что такое виджет?
3. Какой атрибут у кнопки определяет цвет фона?
4. Сколько есть вариантов задания цвета в языке Python?
5. В каких единицах задаются размеры кнопки?

Лабораторная работа № 2 Разработка меню

Пример задания

Для заданной предметной области (например, животный мир) разработать классификацию и представить ее в виде дерева главного меню.

Дополнить две опции главного меню пиктограммами.

Дополнить проект виджетом с фамилией студента в центре формы.

Создать функции показа и сокрытия виджета с фамилией студента.

Привязать разработанные функции к опциям главного меню

Привязать к опциям работы с фамилией студента горячие клавиши.

Разработать контекстное меню для виджета с фамилией.

Привязать контекстное меню к виджету с фамилией.

Контрольные вопросы:

1. Какой атрибут опции меню используется для задания взаимного расположения надписи на опции и пиктограммы
2. Какой виджет используется для привязки пиктограммы к опции меню?
3. К опциям контекстного меню можно привязывать пиктограммы?
4. Какие графические форматы можно использовать для пиктограмм опций меню?
5. Как вызывается контекстное меню?

Лабораторная работа № 3 Разработка усложненного Python-проекта

Пример задания

На форме располагаются компоненты: редактор Entry, группа из четырех кнопок SpeedButton, метка Label, контекстное меню для формы и две кнопки Button (для очистки редактора и закрытия формы) с подходящими по назначению рисунками. В Entry можно ввести только дробное число как аргумент тригонометрической функции. Сама функция: sin, cos, tg или ctg, - выбирается с помощью кнопок SpeedButton. Запуск на вычисление производится с помощью контекстного меню. Результат помещается на Label.

Контрольные вопросы:

1. Какая команда (метод) делает кнопку невидимой?
2. Что задает атрибут bg у кнопки?
3. В какое состояние переходит кнопки, если она нажимается мышью?
4. В каком состоянии находится кнопка исходно?
5. В чем специфика кнопки Speedbutton?

Лабораторная работа №4 Управление временем в Python-проектах

Пример задания

На пространстве формы изображен календарь за январь текущего года с горизонтальным расположением недель. Дни недели подписаны. Квадратная рамка движется циклически по датам с дискретом времени 0.4 сек. Цвет рамки – “black”. Запуск движения – кнопка «Пуск», остановка – кнопка «Стоп», что приводит к установке рамки на первую дату.

Контрольные вопросы:

1. Какой класс позволяет использовать в Python-программе системные часы?
2. Какой класс позволяет использовать в Python-программе системные часы?
3. Какой метод класса datetime используется для работы с системными часами?
4. Как подключаются системные часы к Python-программе?
5. Можно ли при работе со временем в tkinter-приложениях на основе метода after() установить временной интервал меньше одной миллисекунды?

Лабораторная работа №5 Графика на основе канвы

Пример задания

На форме нарисован квадрат со стороной 400 пикселей. Внутри квадрата по периметру движется по часовой стрелке со скоростью 50 пикселей/сек. и шагом в 1 пиксель окружность синего цвета с диаметром 50 пикселей. Управление движением – команды главного меню. Секундная стрелка часов по кругу совершает движение. Пуск/останов движения –кнопка SpeedButton.

Контрольные вопросы:

1. Какой тип графики реализуется виджетом Canvas?
2. Какая фигура создается, если у метода create_arc отсутствует параметр style?
3. В каком виде компьютерной графики можно оперировать отдельным пикселем?
4. Какой параметр задает цвет линии на Canvas?
5. В растровой графике размер графического файла зависит от сложности изображения?

Лабораторная работа № 6 Построение графиков

Пример задания

На координатном пространстве размером 300x300 пикселей отобразить график заданной функции:

$$y = x^2 \sin(\sqrt{x}); \quad 0 \leq x \leq 10; \quad \Delta x = 0.5;$$

График строится в трех вариантах:

- с заданным шагом Δx по оси X;
- с расчетом координаты Y функции для каждого пикселя графика по оси X;
- с использованием сглаживания полилинии 1-го графика.

Контрольные вопросы:

1. Как определяется размах функции?
2. Почему требуется использовать коэффициенты масштабирования при создании графиков?
3. Почему координаты x_t и y_t точки определяются по разным алгоритмам?
4. Какие методы канвы можно использовать для прорисовки графика?
5. Можно ли график сделать разноцветным (точки одного графика разного цвета)?

Лабораторная работа № 7 Обработка матриц

Пример задания

Дана целочисленная матрица $\{A_{ij}\}$ $i=1..n$, $j=1..m$ ($n, m \in (6..10)$). Конкретный размер задается командами главного меню. В правом дополнительном столбце выводится сумма цифр в каждой строке. По нажатию кнопки SpeedButton строка с наименьшей суммой, выделяется красной рамкой.

Контрольные вопросы:

1. Как наиболее быстро очистить канву?
2. Можно ли создать рамку для выделения требуемого элемента матрицы из отдельных отрезков?
3. Какой модуль надо подключить к проекту для генерации случайных чисел?
4. Можно ли сделать матрицу «гребенчатой» (с разным количеством элементов в столбцах)?
5. Как задается диапазон генерируемых случайных чисел?

Лабораторная работа № 8 Обработка событий

Пример задания

Линейка прокрутки определяет сторону квадрата или радиуса окружности в диапазоне 40 \square 80 пикселей. Нажатие левой клавиши мыши приводит к созданию на канве квадрата, правый нижний угол которого находится в месте курсора, а правой - окружности, центр которой находится в месте курсора. Цвет контура фигуры определяется случайно в диапазоне 30000 \square 3000000 и выводится в редакторе Entry. Нажатие клавиши \square или \square приводит к дублированию фигуры со сдвигом на 1 пиксель соответственно влево или вправо.

Контрольные вопросы:

- 1) “<KeyRelease>” – это описание какого события?
- 2) Какое событие является обратным для “<KeyRelease>”?

- 3) Синоним события "<Key>"?
- 4) С какими событиями ассоциируется атрибут num?
- 5) Что можно выяснить через атрибут num события?

Приложение № 3

**ТИПОВЫЕ ЗАДАНИЯ ПО КОНТРОЛЬНОЙ РАБОТЕ
(для студентов заочной формы обучения)**

1. Описать методы и алгоритмы организации выполнения параллельных распределено выполняющихся приложений, реализации взаимодействия параллельных распределенных приложений.
2. Описать программные и аппаратные инструменты исследования методов и алгоритмов реализации взаимодействия параллельных распределенных приложений
3. Охарактеризовать функции библиотеки для реализации алгоритмов организации выполнения параллельных распределено выполняющихся приложений и реализации взаимодействия между ними
4. Определите роль библиотеки MPI при реализации метода параллельных вычислений
5. Какие задачи позволяет решить метод простой итерации решения СЛАУ?
6. В чем сущность функций для скалярного произведения векторов?
7. Опишите CUDA-версии вспомогательных функций
8. В чем заключается суть метода CUDA- реализации?
9. Опишите понятие узла кластера с графическим ускорителем.

Объектом исследования в контрольной работе являются методы и алгоритмы организации выполнения параллельных распределено выполняющихся приложений, реализации взаимодействия параллельных распределенных приложений. Инструментом исследования методов и алгоритмов реализации взаимодействия параллельных распределенных приложений является ЭВМ. Программным средством исследования является библиотека MPI, подключаемая к программным модулям, создаваемым в среде C++. Выполнение работы предусматривает в том числе создание проекта в среде Visual studio, разработку и отладку программы на языке C++ с использованием вызовов требуемых функций библиотеки MPI, запуск программы на выполнение в несколько потоков (требуемое количество копий исходного кода программы), получение результатов работы программы в виде протоколов сообщений (последовательности сообщений от выполняемых программ), комментирующих параллельное выполнение процессов и их взаимодействие в ходе выполнения, получение результатов вычислений.

Приложение № 4

ЗАДАНИЯ И КОНТРОЛЬНЫЕ ВОПРОСЫ ПО КУРСОВОЙ РАБОТЕ

Пример задания

Создать программное приложение «Тренировка внимательности», реализующее следующее задание:

Матрица 10x10. Ячейки матрицы заполнены в случайном порядке изображениями зверей, рыб, растений, людей, строений и т.п. Заполнение матрицы происходит при запуске игры. Одновременно запускается таймер. В отдельном окошке появляются случайно изображения из матрицы. Необходимо в матрице найти аналогичное изображение и щелкнуть по нему мышью, после чего оно исключается из игры. Фиксируются три лучших результата.

Программное приложение дополняется окном со сведениями об авторе разработки, которое вызывается командой главного меню.

Контрольные вопросы к защите:

1. Какую библиотеку необходимо подключить к приложению для того, чтобы обеспечить работу с графическими изображениями?
2. Какой метод позволяет удалить изображение на форме?
3. К какой графике относятся изображения, используемые в тренажере?
4. Как создаются дополнительные окна в tkinter-приложениях?
5. Можно ли для фиксации времени тренировки использовать системные часы компьютера?

Приложение № 5

КОНТРОЛЬНЫЕ ВОПРОСЫ ДЛЯ ЭКЗАМЕНА ПО ДИСЦИПЛИНЕ

Введение в Python. Переменные, константы. Типы данных. Операторы, выражения.

Ввод-вывод данных.

1. Библиотека math.
2. Исключения в Python.
3. Ветвления и циклы в Python.
4. Строки. Базовые операции. Функции и методы строк.
5. Списки. Базовые операции. Функции и методы списков. Генераторы списков.
6. Вложенные списки.
7. Кортежи. Базовые операции.
8. Множества. Базовые операции. Генераторы множеств.
9. Словари. Создание, базовые операции. Методы словарей.
10. Функции. Передача параметров в функцию. Области видимости переменных в Python.
11. Дополнительные возможности при работе с функциями.
12. Функциональные возможности Python (лямбда-функции и т.д.)
13. Хранение структурированных данных в файлах.
14. Модули и пакеты
15. Запуск программ и установка сторонних модулей.
16. Регулярные выражения
17. Работа с файлами и папками в Python.
18. Структура tkinter-программы
19. Окно tkinter-программы
20. Виджет Button: основные свойства
21. Основные события, используемые в tkinter-приложениях
22. Методы привязки функций к виджетам через события
23. Виджет Label: основные свойства
24. Виджет Entry: основные свойства и методы
25. Виджет Text: основные свойства и методы
26. Виджет Listbox: основные свойства и методы
27. Виджет Radiobutton: основные свойства и методы
28. Переменные модуля tkinter
29. Формирование меню в tkinter-приложениях
30. Виджет Scale: основные свойства и методы
31. Виджет Checkbutton: основные свойства и методы
32. Менеджеры разметки
33. Виджет Combobox: основные свойства и методы
34. Виджет Spinbox: основные свойства и методы
35. Виджет Progressbar: основные свойства и методы
36. Кнопка Speedbutton: варианты реализации
37. Виджет Notebook: основные свойства и методы
38. Средства работы со временем в tkinter-приложениях
39. Импорт графики в tkinter-приложения

40. Виджет Canvas: средства формирования статических и динамических изображений

41. События, генерируемые мышью

42. События, генерируемые клавиатурой

43. Свойства событий

44. Работа с текстовыми файлами в tkinter-приложениях

45. Работа с файлами прямого доступа в tkinter-приложениях