

Конференция онлайн по информатике, технологии, робототехнике
Информационно-технологическое образование: от новых компетенций педагога
к формированию конкурентоспособных преимуществ выпускника

Программирование на языке Python и C++: цели и перспективы

Полежаева Ольга Александровна, редактор по информатике
отдела внедрения развивающего обучения и новых продуктов
ООО «Просвещение-Союз» ГК «Просвещение»



Структура пособий

Во время вебинара тут будет окно со спикером

Глава 1 ПРОГРАММИРОВАНИЕ НА ЯЗЫКЕ PYTHON

§ 1 Первые программы

Ключевые слова:

- программа
- транслятор
- линейная программа
- ввод данных
- символьная строка

Что такое программа?

Программа — это набор инструкций, записанных на языке, понятном компьютеру. Компьютер (точнее, его процессор) понимает только один язык — язык машинных кодов, которые записываются в виде цепочек нулей и единиц. Писать программы на таком языке (как делали программисты вычислительных машин первого поколения) очень сложно и долго. Особенно тяжело *отлаживать* их, т. е. находить и исправлять ошибки в таких программах.

Выводы

- Условный оператор **if-else** позволяет выбрать один из двух вариантов действий в зависимости от выполнения некоторого условия.
- Условие, которое нужно проверить, записывается после слова **if**. Записывать обратное условие после **else** не нужно.
- Если условие верно, выполняются все команды, записанные после строки с оператором **if** (со сдвигом вправо).
- Если условие неверно, выполняются все команды, записанные после строки с оператором **else** (со сдвигом вправо).
- В обеих частях условного оператора можно использовать любые операторы, в том числе и другие (вложенные) условные операторы.
- В логической переменной можно хранить логическое значение: **True** (истина) или **False** (ложь).
- Экспертная система — это компьютерная программа, задача которой — заменить человека-эксперта при выработке рекомендаций для принятия решений в сложной ситуации.

Стратегия — это алгоритм игры, который позволяет добиться цели в игре в предположении, что соперники играют безошибочно.



Структура пособий

Во время вебинара тут будет окно со спикером

Интересные сайты

cspstudio.com — программирование для начинающих на C++

ru.cppreference.com — онлайн-справочник по C++

learncs.org — интерактивный учебник по C#

dotnetfiddle.net — онлайн-компилятор C#

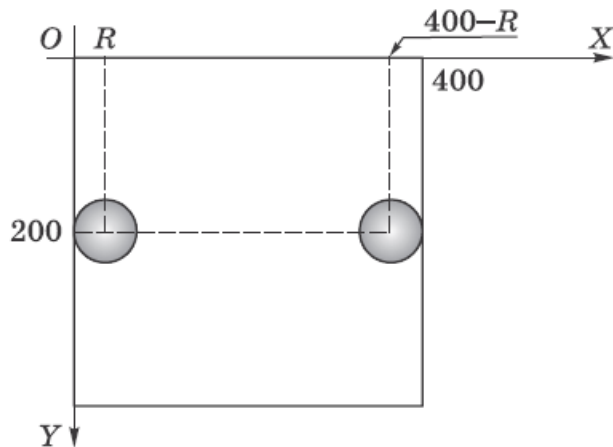
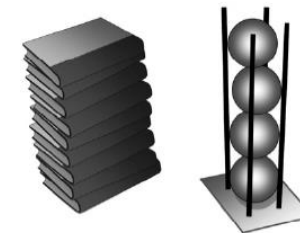


Рис. 2.2

Вопросы и задания

1. Заполните массив случайными целыми числами на отрезке $[0; 4]$ и выведите на экран номера всех элементов, равных значению X (оно вводится с клавиатуры).
2. Алиса начала программу поиска максимального элемента строкой
 $M = A[N-1]$
Закончите программу Алисы (не используя функцию `max`).
3. Елисей начал программу поиска минимального элемента строкой
 $M = A[N//2]$
Закончите программу Елисея (не используя функцию `min`).
4. Объясните, почему при поиске максимального элемента и его номера можно не запоминать само значение максимального элемента.



Содержание части 1

Глава 1. Программирование на языке Python

- § 1. Первые программы
 - Что такое программа?
 - Самая простая программа
 - Вывод текста на экран
- § 2. Диалоговые программы
 - Как тебя зовут?
 - Переменные
 - Сумма чисел
 - Ввод данных в одной строке
- § 3. Компьютерная графика
 - Что такое компьютерная графика?
 - Графика в Python
 - Система координат
 - Управляем пикселями
 - Рисуем линии
 - Прямоугольники
 - Окружность
 - Изменение координат
- § 4. Процедуры
 - Зачем нужны процедуры?
 - Процедура вызывает процедуру
 - Процедуры с параметрами
- § 5. Обработка целых чисел
 - Арифметические выражения
 - Деление нацело
 - Вывод данных на экран
- § 6. Обработка вещественных чисел
 - Что такое вещественное число?
 - Ввод и вывод
 - Операции с вещественными числами
- § 7. Случайные и псевдослучайные числа
 - Случайные и псевдослучайные числа
 - Пишем свой генератор случайных чисел
 - Генератор случайных чисел в Python
- § 8. Ветвления
 - Условный оператор
 - Неполная форма условного оператора
 - Вложенные условные операторы
 - Логические переменные
 - Экспертная система (проект)
- § 9. Сложные условия
 - Операция И
 - Операция ИЛИ
 - Операция НЕ
 - Порядок выполнения операций
- § 10. Циклы с условием
 - Как организовать цикл?
 - Циклы с предусловием
 - Алгоритм Евклида
 - Обработка потока данных
 - Бесконечные циклы
- § 11. Анимация
 - Принципы анимации
 - Начальное положение
 - Анимация движения
 - Обработка нажатия клавиши
- § 12. Циклы по переменной
 - Сделать N раз
 - От цикла `while` к циклу `for`
 - Шаг изменения переменной цикла
- § 13. Циклы в компьютерной графике
 - Узоры
 - Вложенные циклы
 - Рефакторинг
 - Пример
 - Штриховка
 - Штриховка: второй вариант



Во время вебинара тут будет окно со спикером

Глава 2. Программирование на языке C++

- § 14. Первые программы
 - Язык C++
 - Самая простая программа
 - Вывод текста на экран
- § 15. Диалоговые программы
 - Как тебя зовут?
 - Переменные
 - Сумма чисел
- § 16. Компьютерная графика
 - Библиотека `TX Library`
 - Управляем пикселями
 - Линии и фигуры
 - Замкнутые фигуры
- § 17. Процедуры
 - Длинная программа
 - Рефакторинг
 - Процедуры с параметрами
- § 18. Обработка целых чисел
 - Ограниченность значений целых чисел
 - Арифметические выражения
 - Деление и остаток
 - Вывод данных на экран
 - Случайные числа
- § 19. Обработка вещественных чисел
 - Вещественные числа в языке C++
 - Ввод и вывод
 - Операции с вещественными числами
 - Случайные числа
- § 20. Ветвления
 - Условный оператор
 - Вложенные условные операторы
 - Логические переменные
 - Сложные условия
- § 21. Циклы
 - Цикл с предусловием
 - Циклы с постусловием
 - Вычисление квадратного корня
 - Циклы по переменной
- § 22. Анимация
 - Принципы анимации
 - Рисуем шарик
 - Начальное положение
 - Анимация движения
 - Обработка нажатия клавиши

Содержание части 2

Глава 1. Программирование на языке Python

- § 1. Проектирование программ
 - Этапы создания программ
 - Методы проектирования программ
 - Интерфейс и реализация
 - Документирование программы
- § 2. Процедуры
 - Подпрограммы: процедуры и функции
 - Простая процедура
 - Процедуры с параметрами
 - Локальные и глобальные переменные
- § 3. Рекурсия
 - Что такое рекурсия?
 - Ханойские башни
 - Фракталы
- § 4. Функции
 - Что такое функция?
 - Примеры функций
 - Логические функции
 - Рекурсивные функции
- § 5. Символьные строки
 - Что такое символьная строка?
 - Сравнение строк
 - Сложение и умножение
 - Обращение к символам
 - Перебор всех символов
 - Срезы
 - Удаление и вставка
 - Встроенные методы
 - Поиск в символьных строках
 - Замена
 - Преобразования «строка — число»
 - Символьные строки в функциях
 - Рекурсивный перебор
- § 6. Массивы (списки)
 - Что такое массив?
 - Массивы в языке Python
 - Создание массива
 - Обращение к элементу массива
 - Перебор элементов массива
 - Генераторы
 - Вывод массива
 - Ввод массива с клавиатуры
 - Заполнение массива случайными числами
- § 7. Алгоритмы обработки массивов
 - Сумма элементов массива
 - Подсчёт элементов массива, удовлетворяющих условию
 - Особенности копирования списков в Python
- § 8. Поиск в массивах
 - Линейный поиск
 - Поиск максимального элемента в массиве
 - Максимальный элемент, удовлетворяющий условию
- § 9. Используем массивы
 - Игра «Стрельба по тарелкам»
 - Рефакторинг
 - Движение
 - Меняем скорости
 - Бьём тарелки
 - Показываем счёт
- § 10. Матрицы
 - Что такое матрица?
 - Матрицы
 - Вывод матрицы на экран
 - Перебор элементов матрицы
 - Квадратные матрицы
- § 11. Сложность алгоритмов
 - Как сравнивать алгоритмы?
 - Что такое асимптотическая сложность?



Во время вебинара тут будет окно со спикером

Глава 2. Программирование на языке C++

- § 12. Процедуры
 - Простая процедура
 - Процедуры с параметрами
 - Несколько параметров
 - Локальные и глобальные переменные
 - Процедуры, изменяющие аргументы
 - § 13. Рекурсия
 - Рекурсивные процедуры
 - Дерево Пифагора
 - Анимация
 - § 14. Функция
 - Функции в C++
 - Примеры функций
 - Логические функции
 - Рекурсивные функции
 - § 15. Символьные строки
 - Что такое символьная строка?
 - Сравнение строк
 - Сцепление строк
 - Обращение к символам
 - Перебор всех символов
 - Подстрока
 - Удаление и вставка
 - Поиск в символьных строках
 - Замена
 - Преобразования «строка — число»
 - Символьные строки в функциях
 - Рекурсивный перебор
 - § 16. Массивы
 - Массивы в C++
 - Обращение к элементу массива
 - Перебор элементов массива
 - Вывод массива
 - Ввод массива с клавиатуры
 - Заполнение массива случайными числами
 - Алгоритмы обработки массивов
 - § 17. Используем массивы
 - Игра «Стрельба по тарелкам»
 - Движение
 - Бьём тарелки
 - Показываем счёт
 - § 18. Матрицы
 - Что такое матрица?
 - Размещение матрицы в памяти
 - Заполнение матрицы
 - Вывод матрицы на экран
 - Обработка матриц
- ### Управление версиями
- Системы управления версиями
 - Зачем это нужно?
 - Какие бывают системы управления версиями?
 - Основные приёмы работы с Git
 - Начало работы
 - Операции с файлами
 - Не все файлы нужны
 - Восстановление версии
 - Работа с удалённым архивом
 - Ветки
 - Графические оболочки

Содержание части 3

Глава 1. Программирование на языке Python

- § 1. Простые алгоритмы сортировки
 - Что такое сортировка?
 - Метод пузырька (сортировка обменами)
 - Метод выбора
- § 2. Быстрые алгоритмы сортировки
 - Сортировка слиянием
 - Быстрая сортировка
 - Сортировка в языке Python
- § 3. Двоичный поиск
 - Поиск в массиве
 - Что такое двоичный поиск?
 - Двоичный поиск в массиве данных
 - Какой алгоритм поиска лучше?
 - Двоичный поиск по ответу
- § 4. Обработка файлов
 - Какие бывают файлы?
 - Принцип сэндвича
 - Чтение данных
 - Запись данных
 - Вывод файла на экран
 - Суммирование данных из файла
 - Обработка массивов
 - Обработка строк
- § 5. Целочисленные алгоритмы
 - Решето Эратосфена
 - Квадратный корень
- § 6. Словари
 - Что такое словарь?
 - Алфавитно-частотный словарь
 - Перебор элементов словаря
- § 7. Структуры
 - Зачем нужны структуры?
 - Классы
 - Создание структур
 - Что можно делать с полями структуры?
 - Работа с файлами
 - Сортировка
- § 8. Стек, очередь, дек
 - Что такое стек?
 - Использование списка
 - Вычисление арифметических выражений
 - Скобочные выражения
 - Как вызываются подпрограммы?
 - Очередь
 - Дек
- § 9. Деревья
 - Что такое дерево?
 - Деревья поиска
 - Обход дерева
 - Использование связанных структур
 - Вычисление арифметических выражений
 - Модульность
- § 10. Графы
 - Что такое граф?
 - Как описать граф?
 - «Жадные» алгоритмы
 - Минимальное остовное дерево
 - Алгоритм Дейкстры
 - Алгоритм Флойда–Уоршелла
 - Использование списков смежности
- § 11. Динамическое программирование
 - Числа Фибоначчи
 - Что такое динамическое программирование?
 - Кошки и собаки
 - Количество программ для исполнителя
 - Двумерная задача
 - Поиск оптимального решения
- § 12. Игровые модели
 - Выигрышные и проигрышные позиции
 - Кто выиграет?
 - Динамическое программирование
 - Игра с фишками



Во время вебинара тут будет окно со спикером

Глава 2. Программирование на языке C++

- § 13. Простые алгоритмы сортировки
 - Метод пузырька (сортировка обменами)
 - Сортировка вставками
 - Массивы в подпрограммах
- § 14. Быстрые алгоритмы сортировки и поиска
 - Сортировка слиянием
 - Быстрая сортировка
 - Стандартная сортировка в языке C++
 - Двоичный поиск
- § 15. Обработка файлов
 - Принцип сэндвича
 - Файловые потоки
 - Неизвестное количество данных
 - Обработка массивов
 - Чтение файлов по словам
 - Построчная обработка файлов
 - Как передать имя файла программе?
- § 16. Целочисленные алгоритмы
 - Решето Эратосфена
 - «Длинные» числа
- § 17. Динамические массивы и словари
 - Зачем это нужно?
 - Динамические массивы
 - Тип `vector` из библиотеки STL
 - Итераторы
 - Зачем нужны итераторы
 - Словари
 - Перебор элементов словаря
- § 18. Структуры
 - Структуры в C++
 - Обращение к полям структуры
 - Работа с файлами
 - Сортировка
- § 19. Стек, очередь, дек
 - Стек
 - Очередь
 - Хранение очереди в массиве
 - Дек
- § 20. Деревья
 - Деревья в C++
 - Обходы дерева
 - Деревья поиска
 - Вычисление арифметических выражений
 - Дерево в массиве
 - Модульность
- § 21. Графы
 - Графы в языке C++
 - Задача коммивояжера
 - Задача коммивояжера: жадный алгоритм
 - Задача коммивояжера: случайные перестановки
 - Как избавиться от глобальных переменных?
 - Передача данных по ссылке
- § 22. Динамическое программирование
 - Одномерные задачи
 - Редактирование строк
 - Оптимальная стратегия

Содержание части 4

Глава 1. Программирование на языке Python

- § 1. Что такое ООП?
Проблема сложности программ
Процедурное программирование
Объектный подход
Взаимодействие объектов
- § 2. Модель задачи: классы и объекты
Объектно-ориентированный анализ
«Торпедная атака»
Объекты и классы
Классы объектов в игре
Взаимодействие объектов
- § 3. Классы и объекты в программе
Объявление класса
Поля класса
Конструктор
Конструктор с параметрами
Данные класса
Методы
Анимация
Строим флотилию
- § 4. Скрытие внутреннего устройства
Зачем это делать?
Скрытие полей
Доступ к полям через методы
Свойства (property)
Меняем внутреннее устройство
Свойство «только для чтения»
- § 5. Иерархия классов
Наследование
Иерархия объектов в игре
Базовый класс
Доступ к полям
- § 6. Классы-наследники (I)
Классы-наследники
Пульсар
Полиморфизм
- § 7. Классы-наследники (II)
Подвижные объекты
Космические корабли
Модуль с классами
- § 8. Событийно-ориентированное программирование
Особенности современных прикладных программ
Программы с графическим интерфейсом
Простейшая программа
Свойства формы
Обработчик события
- § 9. Использование компонентов (виджетов)
Программа с компонентами
Новый класс: всё в одном
Ввод и вывод данных
Обработка ошибок
- § 10. Создание компонентов
Зачем это нужно?
Компонент для ввода целых чисел
Программа с новым компонентом
Выключатель с рисунком
Перехват щелчка мышью
Новое свойство onChange
Составной компонент
- § 11. Модель и представление
Зачем это нужно?
Вычисление арифметических выражений: модель
Представление



Во время вебинара тут будет окно со спикером

Глава 2. Программирование на языке C++

- § 12. Классы и объекты в C++
Новая задача и её анализ
Класс CMap
Пишем свой конструктор
Рефакторинг
Рисуем карту
- § 13. Программа с классами (практикум)
Класс CCar
Снова рефакторинг
Основная программа
Разбиение на модули
- § 14. Инкапсуляция
Объект защищает свои данные
Изменение внутреннего устройства
Свойство «только для чтения»
Свойства в C#
- § 15. Наследование
Моделирование жизни в океане
Иерархия классов
Базовый класс
Абстрактный класс
Защищённые поля и методы (protected)
Неподвижные объекты
Подвижные объекты
Рыбы
Хищники
Вспомогательные процедуры и функции
Основная программа
- § 16. Полиморфизм
Указатели на базовый класс
Полиморфизм в действии
Немного теории
Позднее связывание
Класс Ocean
Деструктор
- § 17. Взаимодействие объектов
Столкновения
Изменение базового класса
Изменение других классов
«Умные» указатели
- § 18. Простая программа на C#
RAD-среды для разработки программ
Язык C# и среда .NET
Проект в C#
Свойства объектов
Обработчики событий
- § 19. Использование компонентов
Программа для просмотра рисунков
Ввод и вывод данных
Обработка ошибок
- § 20. Создание новых классов
Вычисление арифметических выражений: модель
Новый класс
Методы класса
Вычисление арифметических выражений: представление
Новый компонент — зачем это нужно?
Добавляем новый компонент
Изменение поведения компонента

Преимущества пособия

Во время вебинара тут будет окно со спикером

- Изучение сразу двух современных языков программирования. Умение программировать на разных языках – обязательное требование к разработчику программного обеспечения
- Вся глубина сложных фундаментальных понятий и приёмов программирования передаётся доступным для школьника языком
- Эффективные приёмы развития алгоритмического мышления и мышления вообще
- Профориентация для будущих программистов
- Подготовка к дальнейшему обучению в вузах, готовящих IT-специалистов
- Разбор и решение задач из пособия снимает многие затруднения при решении выпускниками задач ЕГЭ (часть С)

От простого – к сложному

Во время вебинара тут будет окно со спикером

Сумма двух чисел

```
num1 = int( input() )
num2 = int( input() )
summa = num1 + num2
print( summa )
```

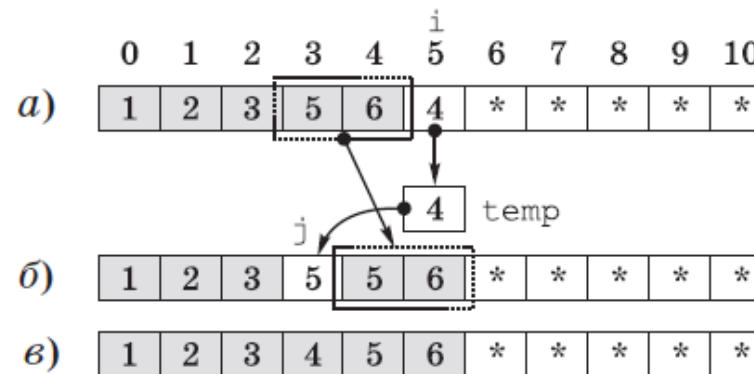
Цикл в стиле Python

```
B = [x for x in A
     if x % 2 == 0]
print( len(B) )
```

Сортировка методом пузырька

```
for( int i = 0; i < N-1; i++ )
    for( int j = N-2; j >= i; j-- )
        if ( A[j] > A[j+1] ) {
            int temp = A[j];
            A[j] = A[j+1];
            A[j+1] = temp;
        }
```

Сортировка вставками



Написание программ вместе с автором

Во время вебинара тут будет окно со спикером

В нашей игре с камнями позиция задаётся одним целым числом — количеством оставшихся камней, его и нужно передать функции как параметр:

```
def isWinPos( N ):
    ...
```

По условию игра заканчивается, когда камней не остаётся, поэтому можно написать логическую функцию, определяющую, завершена ли игра:

```
def gameOver( N ):
    return (N <= 0)
```

Согласно условию, игрок выиграл, если соперник закончил игру (взял последний камень и проиграл), поэтому в этом случае функция `isWinPos` сразу вернёт значение `True`:

```
def isWinPos( N ):
    if gameOver( N ):
        return True
    ...
```

Если же игра ещё не завершена (камни остались), нужно проверить все возможные ходы: в этой игре мы можем уменьшать количество камней на один или на два. Если хотя бы один из этих ходов ведёт в проигрышную позицию, то данная позиция выигрышная. Попад в неё, соперник обязательно проиграет. Если же оба хода ведут в выигрышные (для соперника) позиции, то позиция проигрышная и функция должна вернуть значение `False`:

```
def isWinPos( N ):
    if gameOver( N ):
        return True
    for take in range(1, 3):
        if not isWinPos( N-take ):
            return True
    return False
```


Трассировочные таблицы

Во время вебинара тут будет окно со спикером

	Оператор	i	summa
1	summa = 0		0
2	i = 0	0	
3	summa += A[0]		5
4	i += 1	1	
5	summa += A[1]		7
6	i += 1	2	
7	summa += A[2]		15
8	i += 1	3	
9	summa += A[3]		18
10	i += 1	4	
11	summa += A[4]		19

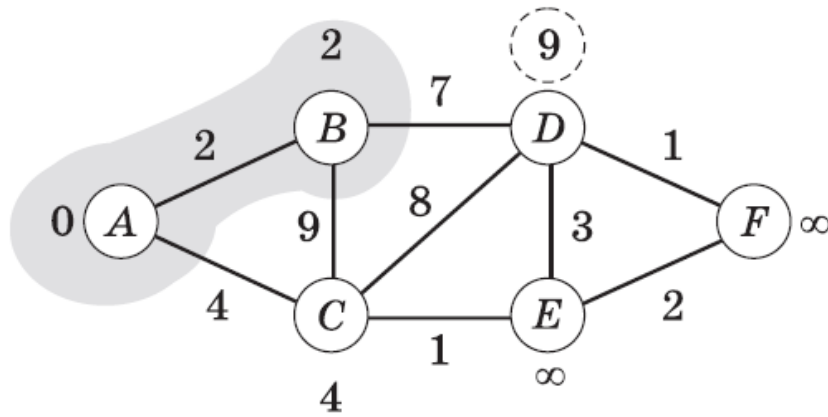
```
summa = 0
for i in range(N):
    summa += A[i]
print( summa )
```

	0	1	2	3	4
A	5	2	8	3	1

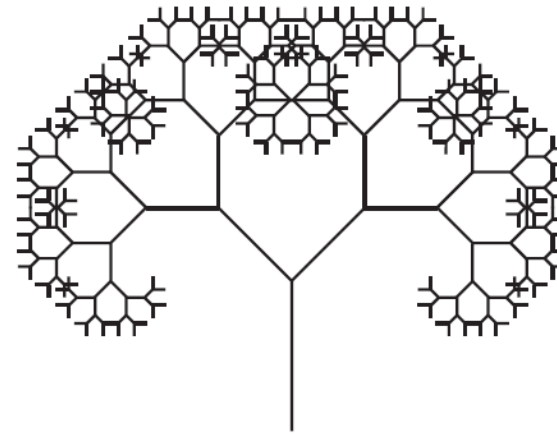
Примеры задач

Во время вебинара тут будет окно со спикером

Кратчайший путь в графе



Рекурсия

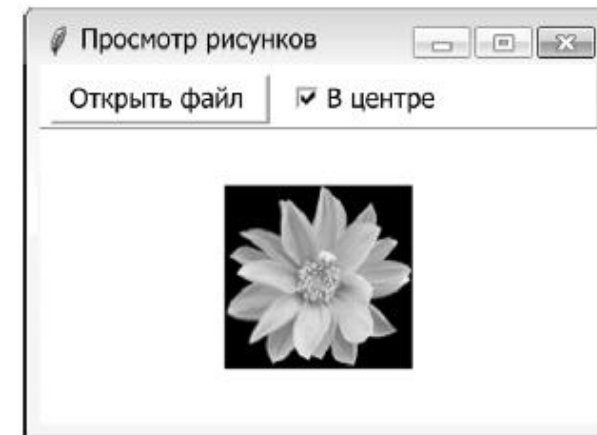
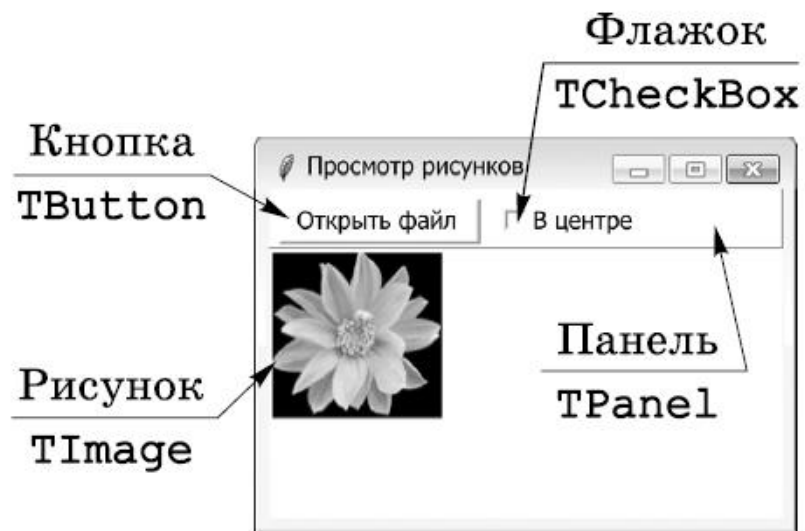
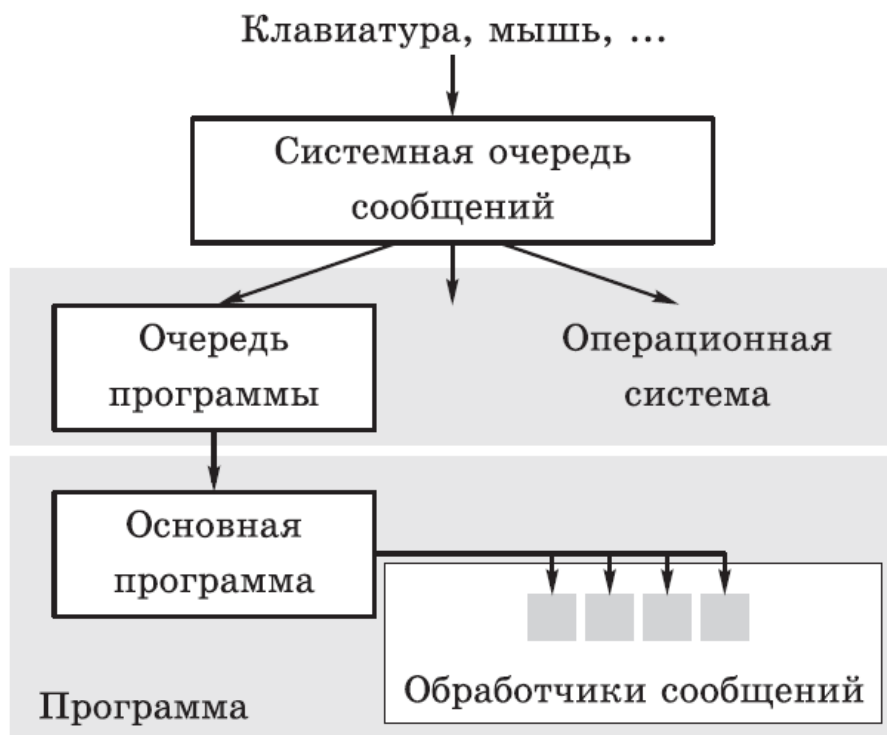


Выбор оптимального решения

Задача о розливе молока. В цистерне N литров молока. Есть бидоны объёмом 1, 5 и 6 литров. Нужно разлить молоко в бидоны так, чтобы все используемые бидоны были заполнены и их количество было минимальным.

Событийно-ориентированное программирование

Во время вебинара тут будет окно со спикером



Моделирование жизни в океане

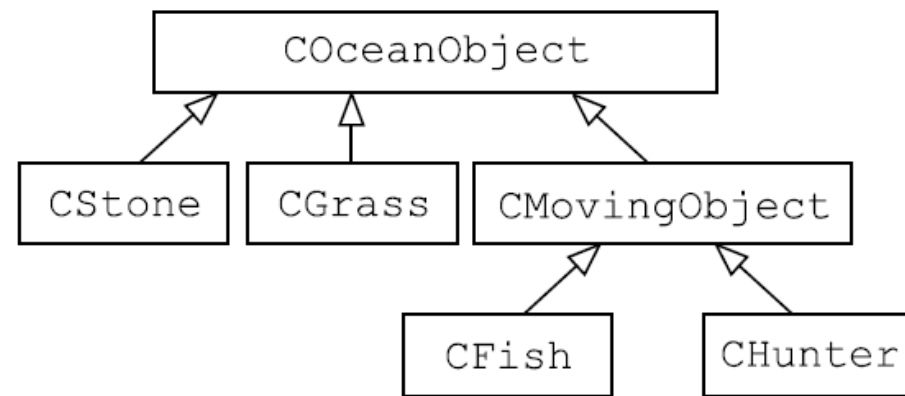
В этом параграфе мы начнём разрабатывать новую игру, которая моделирует жизнь в океане. На игровом поле «живут» разнообразные объекты:

- *Камни* (препятствия) — не двигаются;
- *Трава* (планктон) — пища для рыб;
- *Рыбы*, поедающие траву; они не обращают внимания на камни;
- *Хищники* — хищные рыбы, поедающие обычных *Рыб*; они не обращают внимания на камни и траву.

Океан живёт своей жизнью. При этом происходят следующие изменения¹⁾:

- *Трава* растёт (через заданное время её размеры увеличиваются);
- *Рыбы* двигаются в выбранном направлении (или случайно меняют направление движения);
- если *Рыба* проходит через *Траву*, то размер *Травы* уменьшается, а *Рыба* становится сытой;
- если *Рыба* долгое время не ест, она умирает от голода;
- если *Хищник* сталкивается с *Рыбой*, он ест *Рыбу*; при этом *Рыба* погибает, а *Хищник* становится сытым;
- если *Хищник* долгое время не ест, он умирает от голода.

Наша первая (и самая важная) задача — построить систему классов для этой игры.



Сайт поддержки!

Во время вебинара тут будет окно со спикером

<http://kpolyakov.spb.ru/school/русcpp.htm>

The screenshot shows the website interface with a navigation bar containing links for 'главная', 'школа', 'вуз', 'наука', 'delphi', 'программы', 'походы', and 'автор'. A search bar with the Google logo is present. The main content area features a section titled 'Учебное пособие «Программирование. 8-11 классы. Профильная школа»' with social media links for 'Форум', 'контакте', and 'Telegram'. Below this is a 'Что это такое?' section with introductory text. A sidebar on the left lists various resources like 'Программа', 'Учебник 7-9', 'Учебник 10-11(Б+У)', and 'Учебник 10-11(У)'. A QR code is visible at the bottom right of the main content area.

ЕГЭ
Подготовка к ЕГЭ по информатике.

Презентации для проведения уроков информатики.

ЛамПанель
Учебная модель компьютера для программирования на машинном языке.

Иванов 5
Петров 4
Сидоров 3
Компьютерное тестирование знаний в локальной сети — программа «NetTest». Просто и удобно.

The screenshot shows the website interface with a navigation bar containing links for 'главная', 'школа', and 'вуз'. The main content area features a section titled 'Учебник 7-9' with a list of resources: 'Материалы', 'Тесты', 'Практикум', 'Презентации', 'Python', and 'C++'. Below this are links for 'Учебник 10-11(Б+У)' and 'Учебник 10-11(У)'. The sidebar on the left lists various resources like 'Программа', 'Учебник 7-9', 'Учебник 10-11(Б+У)', and 'Учебник 10-11(У)'.

Во время вебинара тут будет окно со спикером

Интернет- магазины:

www.shop.prosv.ru

www.labirint.ru

www.my-shop.ru

Группа компаний «Просвещение»

Адрес: 127473, г. Москва,
ул. Краснопролетарская, д. 16, стр. 3,
подъезд 8, бизнес-центр «Новослободский»

Горячая линия: vopros@prosv.ru