

**Муниципальное общеобразовательное бюджетное Учреждение Лицей №6
муниципального района Мелеузовский район Республики Башкортостан**

РАССМОТРЕНО

на заседании методической
кафедры (объединения)
Протокол №1 от 25.08.2023г.
Руководитель кафедры
(объединения)

 /Рыбина С.А.
(ФИО руководителя)

СОГЛАСОВАНО

Заместитель директора по ВР
 / С.А. Рыбина

УТВЕРЖДЕНО

приказом директора
МОБУ Лицей № 6
от «30» августа 2023г. №202

Рабочая программа

курса внеурочной деятельности

«Программирование на языке Python, C++»

Уровень - среднее общее образование

Классы – 10-11

Срок реализации - 1 год

2023 г.

Содержание программы

Первый уровень

Программирование на языке Python (18 часов).

Сортировка массивов. Метод пузырька (сортировка обменами). Метод выбора. Сортировка слиянием. Быстрая сортировка. Сортировка в языке Python.

Двоичный поиск в массиве данных. Двоичный поиск по ответу.

Обработка файлов. Типы файлов. Чтение данных. Запись данных. Обработка данных из файла.

Целочисленные алгоритмы. Решето Эратосфена. Целочисленный квадратный корень.

Словари. Алфавитно-частотный словарь. Перебор элементов словаря.

Структуры. Классы. Создание структур. Работа с полями структур. Хранение структур в файлах. Сортировка структур.

Стек. Использование списка. Вычисление арифметических выражений. Скобочные выражения. Системный стек. Очередь. Дек.

Деревья. Деревья поиска. Обход дерева. Использование связанных структур. Вычисление арифметических выражений.

Графы. Описание графа. Жадные алгоритмы. Минимальное остовное дерево. Алгоритм Дейкстры. Алгоритм Флойда–Уоршелла. Использование списков смежности.

Динамическое программирование. Числа Фибоначчи. Количество программ для исполнителя. Двумерные задачи. Поиск оптимального решения.

Игровые модели. Выигрышные и проигрышные позиции.

Программирование на языке C++ (14 часов).

Сортировка массивов. Метод пузырька (сортировка простыми обменами). Сортировка вставками. Массивы в подпрограммах. Сортировка слиянием. Быстрая сортировка. Стандартная сортировка в языке C++. Двоичный поиск.

Обработка файлов. Файловые потоки. Обработка данных из файла. Чтение текстовых файлов по словам. Построчная обработка файлов. Аргументы основной программы.

Целочисленные алгоритмы. Решето Эратосфена. «Длинные» числа.

Динамические массивы. Тип `vector` из библиотеки STL. Итераторы.

Словари. Перебор элементов словаря.

Структуры в C++. Обращение к полям структуры. Хранение структур в файлах. Сортировка структур.

Стек. Очередь. Хранение очереди в массиве. Дек.

Деревья в C++. Обходы дерева. Деревья поиска. Вычисление арифметических выражений. Хранение дерева в массиве.

Графы в языке C++. Задача коммивояжёра. Жадные алгоритмы. Случайные перестановки. Передача данных по ссылке.

Динамическое программирование. Одномерные задачи. Редактирование строк. Оптимальная стратегия.

Резерв – 2 часа.

Второй уровень

Программирование на языке Python (18 часов). Проблема сложности программ. Процедурный и объектно-ориентированный подходы к написанию программ.

Классы и объекты. Объектно-ориентированный анализ. Взаимодействие объектов. Свойства и методы.

Классы и объекты в программе. Объявление класса. Поля класса. Конструктор класса. Данные и методы класса.

Скрытие внутреннего устройства. Доступ к полям через методы. Свойства (*property*). Свойство «только для чтения»

Иерархия классов. Наследование. Базовый класс. Доступ к полям. Классы-наследники. Полиморфизм. Разработка модулей.

Событийно-ориентированное программирование. Программы с графическим интерфейсом. Форма. Свойства формы. Обработчики событий.

Использование компонентов (виджетов). Ввод и вывод данных. Обработка ошибок с помощью исключений.

Создание компонентов. Добавление свойств и методов. Составные компоненты.

Модель и представление.

Программирование на языке C++ (14 часов). Классы и объекты в языке C++. Объектно-ориентированный анализ задачи. Конструкторы классов. Разбиение на модули.

Инкапсуляция. Возможность изменения внутреннего устройства объектов. Свойства «только для чтения».

Наследование. Иерархия классов. Базовый класс. Абстрактный класс. «Чистые» виртуальные методы. Защищённые поля и методы (*protected*).

Полиморфизм. Указатели на базовый класс. Виртуальные методы. Позднее связывание. Деструктор.

Организация взаимодействия объектов.

«Умные» указатели.

RAD-среды для разработки программ. Язык C# и среда .NET. Проект в C#. Свойства объектов. Обработчики событий.

Использование компонентов. Ввод и вывод данных. Обработка ошибок с помощью исключений.

Создание новых классов. Статические методы класса. Создание новых компонентов.

Резерв – 2 часа.

Планируемые результаты

Первый уровень

В результате изучения курса на третьем уровне учащийся

- 1) научиться применять различные алгоритмы сортировки массивов;
- 2) научиться использовать двоичный поиск;
- 3) научиться обрабатывать данные, записанные в текстовые и двоичные файлы, и сохранять в файлах результаты работы программы;
- 4) научиться использовать структуры для объединения данных;
- 5) научиться применять словари, стеки, очереди, деки для решения задач обработки данных;
- 6) научиться использовать деревья для организации данных;
- 7) познакомиться с методами описания графов и некоторыми популярными алгоритмами на графах;
- 8) научиться использовать динамическое программирование для решения комбинаторных и оптимизационных задач;
- 9) познакомиться с понятием выигрышных и проигрышных позиций в играх с полной информацией;

Второй уровень

В результате изучения курса на четвёртом уровне учащийся

- 10) познакомиться с объектно-ориентированным подходом к разработке программ;
- 11) научиться выполнять объектно-ориентированный анализ задачи, выделять свойства и методы объектов;
- 12) научиться использовать инкапсуляцию для защиты данных объектов;
- 13) познакомиться с понятиями «класс» и «абстрактный класс»;
- 14) познакомиться с понятиями «инкапсуляция», «наследование», «полиморфизм»;
- 15) научиться проектировать несложные иерархии классов для прикладных задач;

- 16) познакомиться с принципами разработки событийно-ориентированных программ;
- 17) научиться создавать программы с графическим интерфейсом на языках Python и C#;
- 18) научиться использовать готовые и создавать новые компоненты (виджеты) для сред быстрой разработки программ.

Поурочное планирование курса
1 уровень (34 часа)

Таблица 1.

Номер урока	Тема занятия	Параграф пособия (номер, название)	Кол-во часов	
			теория	практи- ка
	Программирование на языке Python			
1.	Простые алгоритмы сортировки	§ 1. Простые алгоритмы сортировки	0,5	0,5
2.	Сортировка слиянием	§ 2. Быстрые алгоритмы сортировки	0,5	0,5
3.	Быстрая сортировка	§ 2. Быстрые алгоритмы сортировки	0,5	0,5
4.	Двоичный поиск	§ 3. Двоичный поиск	0,5	0,5
5.	Обработка файлов	§ 4. Обработка файлов	0,5	0,5
6.	Обработка файлов: практикум	§ 4. Обработка файлов		1
7.	Целочисленные алгоритмы	§ 5. Целочисленные алгоритмы	0,5	0,5
8.	Словари	§ 6. Словари	0,5	0,5
9.	Структуры	§ 7. Структуры	0,5	0,5
10.	Структуры: практикум	§ 7. Структуры		1
11.	Стек, очередь, дек	§ 8. Стек, очередь, дек	0,5	0,5
12.	Деревья	§ 9. Деревья	0,5	0,5
13.	Графы	§ 10. Графы	0,5	0,5
14.	Графы: практикум	§ 10. Графы		1
15.	Динамическое программирование	§ 11. Динамическое программирование	0,5	0,5
16.	Динамическое программирование: практикум	§ 11. Динамическое программирование	0,5	0,5
17.	Игровые модели	§ 12. Игровые модели	0,5	0,5
18.	Игровые модели: практикум	§ 12. Игровые модели		1
	Программирование на языке C++			
19.	Простые алгоритмы сортировки	§ 13. Простые алгоритмы сортировки	0,5	0,5
20.	Быстрые алгоритмы сортировки и поиска	§ 14. Быстрые алгоритмы сортировки и поиска	0,5	0,5
21.	Обработка файлов	§ 15. Обработка файлов	0,5	0,5

Номер урока	Тема занятия	Параграф пособия (номер, название)	Кол-во часов	
			теория	практи- ка
22.	Целочисленные алгоритмы	§ 16. Целочисленные алгоритмы	0,5	0,5
23.	Динамические массивы и словари	§ 17. Динамические массивы и словари	0,5	0,5
24.	Итераторы	§ 17. Динамические массивы и словари	0,5	0,5
25.	Структуры	§ 18. Структуры	0,5	0,5
26.	Структуры: практикум	§ 18. Структуры		1
27.	Стек, очередь, дек	§ 19. Стек, очередь, дек	0,5	0,5
28.	Деревья	§ 20. Деревья	0,5	0,5
29.	Графы	§ 21. Графы	0,5	0,5
30.	Графы: практикум	§ 21. Графы		1
31.	Динамическое программирование	§ 22. Динамическое программирование	0,5	0,5
32.	Динамическое программирование: практикум	§ 22. Динамическое программирование		1
33.	Резерв			1
34.	Резерв			1
		Итого	11,5	22,5

2 уровень (34 часа)

Таблица 2.

Номер урока	Тема занятия	Параграф пособия (номер, название)	Кол-во часов	
			теория	практи- ка
	Программирование на языке Python			
1.	Что такое ООП?	§ 1. Что такое ООП?	1	
2.	Модель задачи: классы и объекты	§ 2. Модель задачи: классы и объекты	0,5	0,5
3.	Классы и объекты в программе	§ 3. Классы и объекты в программе	0,5	0,5
4.	Классы и объекты в программе: практикум	§ 3. Классы и объекты в программе		1
5.	Скрытие внутреннего устройства	§ 4. Скрытие внутреннего устройства	0,5	0,5

Номер урока	Тема занятия	Параграф пособия (номер, название)	Кол-во часов	
			теория	практи- ка
6.	Иерархия классов	§ 5. Иерархия классов	0,5	0,5
7.	Классы-наследники (I)	§ 6. Классы-наследники (I)	0,5	0,5
8.	Классы-наследники (II)	§ 7. Классы-наследники (II)	0,5	0,5
9.	Доработка игры	§ 5-7.		1
10.	Событийно-ориентированное программирование	§ 8. Событийно-ориентированное программирование	0,5	0,5
11.	Использование компонентов (виджетов)	§ 9. Использование компонентов (виджетов)	0,5	0,5
12.	Использование компонентов (виджетов)	§ 9. Использование компонентов (виджетов)	0,5	0,5
13.	Создание компонентов	§ 10. Создание компонентов	0,5	0,5
14.	Модель и представление	§ 11. Модель и представление	0,5	0,5
15.	Выполнение проекта	§ 1-11.		1
16.	Выполнение проекта	§ 1-11.		1
	Программирование на языке C++			
17.	Классы и объекты	§ 12. Классы и объекты	0,5	0,5
18.	Программа с классами (практикум)	§ 13. Программа с классами (практикум)	0,5	0,5
19.	Программа с классами (практикум)	§ 13. Программа с классами (практикум)		1
20.	Инкапсуляция	§ 14. Инкапсуляция	0,5	0,5
21.	Наследование	§ 15. Наследование	0,5	0,5
22.	Наследование: практикум	§ 15. Наследование		1
23.	Полиморфизм	§ 16. Полиморфизм	0,5	0,5
24.	Полиморфизм: практикум	§ 16. Полиморфизм		1
25.	Взаимодействие объектов	§ 17. Взаимодействие объектов	0,5	0,5
26.	Простая программа на C#	§ 18. Простая программа на C#	0,5	0,5
27.	Использование компонентов	§ 19. Использование компонентов	0,5	0,5

Номер урока	Тема занятия	Параграф пособия (номер, название)	Кол-во часов	
			теория	практи- ка
28.	Ввод и вывод данных	§ 19. Использование компонентов	0,5	0,5
29.	Создание новых классов	§ 20. Создание новых классов	0,5	0,5
30.	Выполнение проекта	§ 1-20.		1
31.	Выполнение проекта	§ 1-20.		1
32.	Выполнение проекта	§ 1-20.		1
33.	Резерв			1
34.	Резерв			1
		Итого	11,5	22,5

Оценивание результатов обучения

Результатом обучения считается способность учащегося написать программу (разработать проект) определённого уровня сложности. Далее выделяются следующие уровни сложности:

- А:** начальный уровень, воспроизведение изучаемого материала с незначительными изменениями;
- В:** средний уровень, способно применять изученный материал для написания программ, которые отличаются от изученных;
- С:** высокий уровень, способно применять изученный материал для самостоятельного написания программ, решающих нестандартные задачи.

Первый уровень

Глава 1. Программирование на языке Python

§ 1. Простые алгоритмы сортировки

- А:** Напишите программу, которая выполняет сортировку массива методом «пузырька».
- В:** Напишите программу, которая выполняет сортировку массива методом «пузырька» в обратном порядке (сверху вниз).
- С:** Напишите программу, которая выполняет сортировку двух половин массива (отдельно) методом выбора.

§ 2. Быстрые алгоритмы сортировки

- А:** Напишите программу, которая выполняет сортировку массива по убыванию методом слияния.
- В:** Напишите программу, которая выполняет быструю сортировку массива символьных строк по убыванию длины.
- С:** Напишите программу, которая сравнивает количество операций при сортировке массива методами «пузырька» и слияния.

§ 3. Двоичный поиск

- А:** Напишите программу, которая находит в отсортированном массиве индексы всех элементов, равных заданному значению X .
- В:** Напишите программу, которая сравнивает количество проверок при линейном и двоичном поиске в отсортированном массиве.
- С:** Напишите программу, которая определяет среднее количество проверок при двоичном поиске в отсортированном массиве. Используя результаты работы этой программы, постройте зависимости количества проверок от размера массива.

§ 4. Обработка файлов

- А:** Напишите программу, которая вычисляет сумму чисел, записанных в файле в столбик.

В: Напишите программу, которая сортирует числа, записанные в файле в столбик.

С: Напишите программу, которая сортирует набор чисел, записанных в двух файлах в столбик. Отсортированные числа должны быть записаны в новый файл.

§ 5. Целочисленные алгоритмы

А: Напишите программу, которая вводит натуральное число N и находит все простые числа на отрезке $[2; N]$. Используйте алгоритм «решето Эратосфена».

В: Напишите программу, которая вводит натуральные числа M и N и находит все простые числа на отрезке $[M; N]$. Используйте алгоритм «решето Эратосфена».

С: Напишите программу, которая определяет количество операций, которые выполняются при решении задачи A . Используя результаты работы этой программы, постройте зависимость количества операций (или времени работы программы) от N .

§ 6. Словари

А: Напишите программу, которая строит алфавитно-частотный словарь для заданного файла, в котором каждое слово записано в отдельной строке.

В: Напишите программу, которая строит алфавитно-частотный словарь для заданного файла с произвольным текстом.

С: Напишите программу, которая сравнивает тексты разных авторов на основе частоты использования предлогов и союзов.

§ 7. Структуры

А: Напишите программу управления базой данных. Она должна поддерживать добавление, удаление и просмотр записей.

В: Напишите программу управления базой данных. Она должна поддерживать добавление, удаление, просмотр и поиск записей по ключу.

С: Напишите программу управления базой данных. Она должна поддерживать добавление, удаление, просмотр, поиск записей по ключу и сортировку по ключу.

§ 8. Стек, очередь, дек

А: Напишите программу, которая проверяет правильность скобочного выражения с тремя типами скобок.

В: Напишите программу, которая вычисляет значение арифметического выражения, записанного в префиксной форме.

С: Напишите программу, которая применяет алгоритм заливки области для поиска пути в лабиринте.

§ 9. Деревья

- А:** Напишите программу, которая вычисляет введенное арифметическое выражение без скобок.
- В:** Напишите программу, которая вычисляет введенное арифметическое выражение со скобками.
- С:** Напишите программу, которая вычисляет введенное арифметическое выражение со скобками и функциями (*sin*, *cos*, *sqrt*, *abs*).

§ 10. Графы

- А:** Напишите программу, которая строит минимальное остовное дерево для неориентированного графа.
- В:** Напишите программу, которая определяет кратчайший маршрут из одной вершины графа в другую с помощью алгоритма Дейкстры.
- С:** Напишите программу, которая определяет кратчайшие маршруты между каждой парой вершин графа с помощью алгоритма Флойда-Уоршелла.

§ 11. Динамическое программирование

- А:** Напишите программу, которая вычисляет количество битовых цепочек длины N , в которых нет двух единиц подряд.
- В:** Напишите программу, которая вычисляет количество битовых цепочек длины N , в которых нет трёх единиц подряд.
- С:** Напишите программу, которая решает задачу о ранце.

§ 12. Игровые модели

- А:** Напишите программу, которая играет с человеком в игру Баше.
- В:** Напишите программу, которая определяет тип позиции (выигрышная или проигрышная) в игре «крестики-нолики» на доске 3×3 .
- С:** Напишите программу, которая играет с человеком в «крестики-нолики».

Глава 2. Программирование на языке C++

§ 13. Простые алгоритмы сортировки

- А:** Напишите программу, которая сортирует массив по убыванию методом вставок.
- В:** Напишите программу, которая сортирует массив по убыванию суммы цифр методом вставок.
- С:** Напишите программу, которая сравнивает время сортировки массивов разных размеров методом вставок и методом бинарных вставок.

§ 14. Быстрые алгоритмы сортировки и поиска

- А:** Напишите программу, которая выполняет быструю сортировку массива по убыванию и определяет количество выполненных обменов.
- В:** Напишите программу, которая определяет среднее количество обменов при быстрой сортировке массивов разной длины.

С: Напишите программу, которая сравнивает время быстрой сортировки массива в вашей реализации и время работы стандартной функции сортировки.

§ 15. Обработка файлов

А: В файле записаны целые числа. Напишите программу, которая выводит в другой файл все чётные числа, содержащиеся в исходном файле.

В: В файле записаны целые числа. Напишите программу, которая выводит в другой файл все простые числа, содержащиеся в исходном файле.

С: В файле записаны текст, содержащий целые числа. Напишите программу, которая выводит в другой файл все числа Фибоначчи, содержащиеся в исходном файле.

§ 16. Целочисленные алгоритмы

А: Напишите программу, которая вычисляет все цифры десятичной записи числа 2^{100} .

В: Напишите программу, которая вводит из файла два «длинных» числа и выводит в другой файл их сумму.

С: Напишите программу, которая вводит из файла два «длинных» числа и выводит в другой файл их произведение.

§ 17. Динамические массивы и словари

А: Напишите программу, которая вводит натуральное число N и выводит все простые числа на отрезке $[2; N]$. Используйте алгоритм «решето Эратосфена».

В: Напишите программу, которая строит алфавитно-частотный словарь для файла, в котором каждое слово записано в отдельной строке.

С: Напишите программу, которая строит алфавитно-частотный словарь для файла, содержащего произвольный текст.

§ 18. Структуры

А: Напишите программу, которая управляет базой данных, хранящейся в двоичном файле. Программа должна обеспечивать добавление, удаление и просмотр записей.

В: Напишите программу, которая управляет базой данных, хранящейся в двоичном файле. Программа должна обеспечивать добавление, удаление, просмотр и поиск записей по ключу.

С: Напишите программу, которая управляет базой данных, хранящейся в двоичном файле. Программа должна обеспечивать добавление, удаление, просмотр, поиск и сортировку записей по ключу. Команды должны задаваться в текстовом виде (как в языке SQL).

§ 19. Стек, очередь, дек

А: Напишите программу, которая с помощью стека вычисляет значение арифметического выражения, записанного в постфиксной форме.

В: Напишите программу, которая с помощью стека вычисляет значение арифметического выражения, записанного в постфиксной форме. В выражении можно использовать вызовы функций *sin*, *cos*, *abs*, *sqrt*.

С: Напишите программу, которая переводит инфиксную запись арифметического выражения в постфиксную форму.

§ 20. Деревья

А: Напишите программу, которая строит двоичное дерево поиска на основе данных из файла и выводит отсортированную последовательность чисел.

В: Напишите программу, которая вводит арифметическое выражение в виде символьной строки и вычисляет его значение с помощью дерева. Выражение может содержать скобки.

С: Напишите программу, которая вводит арифметическое выражение в виде символьной строки и вычисляет его значение с помощью дерева. Выражение может содержать скобки и вызовы функций *sin*, *cos*, *abs*, *sqrt*.

§ 21. Графы

А: Напишите программу, которая решает задачу коммивояжёра с помощью жадного алгоритма.

В: Напишите программу, которая решает задачу коммивояжёра с помощью случайных перестановок.

С: Напишите программу, которая решает задачу коммивояжёра с помощью случайных перестановок. Сравните различные способы случайных перестановок.

§ 22. Динамическое программирование

А: Напишите программу, которая находит программу минимальной длины для исполнителя Калькулятор (см. текст параграфа).

В: Напишите программу, которая находит расстояние Левенштейна между двумя символьными строками.

С: Напишите программу, которая находит расстояние Левенштейна между двумя символьными строками с учётом различной стоимости операций.

Второй уровень

Глава 1. Программирование на языке Python

§ 1. Что такое ООП?

§ 2. Модель задачи: классы и объекты

А: Перечислите свойства и методы, которыми могли бы обладать объекты класса *Яхта*.

В: Постройте объектно-ориентированную модель для задачи моделирования дорожного движения. Рассмотрите классы объектов *Дорога*, *Светофор*, *Машина* и их взаимосвязь.

С: Постройте объектно-ориентированную модель для придуманной вами задачи моделирования.

§ 3. Классы и объекты в программе

А: Закончите программу, рассмотренную в параграфе.

В: Добавьте в программу из параграфа класс *Торпедный аппарат*.

С: Добавьте в программу из параграфа классы *Торпедный аппарат* и *Торпеда*.

§ 4. Скрытие внутреннего устройства

А: Постройте программную реализацию класса *Перо*. Все данные должны быть скрыты.

В: Постройте программную реализацию класса *Машина*. Все данные должны быть скрыты.

С: Измените программу игры «Торпедная атака» так, чтобы все поля у объектов были закрытыми.

§ 5. Иерархия классов

А: В программе нужно моделировать следующие классы: *Озеро*, *Пресное озеро*, *Солёное озеро*, *Река*, *Море*, *Водоём*, *Океан*. Постройте иерархию классов для этой задачи.

В: В программе нужно моделировать следующие классы: *Дерево*, *Осина*, *Берёза*, *Озеро*, *Река*, *Заяц*, *Волк*, *Медведь*. Постройте иерархию классов для этой задачи.

С: В программе нужно моделировать следующие классы: *Корабль*, *Подводная лодка*, *Самолет*, *Вертолет*, *Гидросамолет*, *Мотоцикл*, *Трактор*. Постройте иерархию классов для этой задачи.

§ 6. Классы-наследники (I)

А: Закончите программу из параграфа.

В: Добавьте в программу из задания *A* ещё один класс неподвижных объектов, которые изображаются квадратом или ромбом.

С: Добавьте в программу из задания *A* ещё один класс неподвижных объектов, которые при анимации изменяются случайным образом.

§ 7. Классы-наследники (II)

А: Закончите программу из текста параграфа.

В: Измените программу так, чтобы космический корабль, который попал в чёрную дыру или в пульсар, уничтожался. Вместо него должен появляться новый корабль в случайном месте.

С: Добавьте в игру объекты класса *TDestroyer* – специальные боевые космические корабли, которые охотятся за кораблями и странниками и уничтожают их при встрече.

§ 8. Событийно-ориентированное программирование

А: Постройте программу, которая запрашивает разрешение на завершение работы.

В: Измените цвет формы в программе из задания *А*.

С: Доработайте программу из задания *А* так, чтобы при щелчке мышью по форме изменялся цвет формы.

§ 9. Использование компонентов (виджетов)

А: Закончите программу для работы с RGB-кодами цвета.

В: Напишите программу для перевода чисел из восьмеричной системы счисления в десятичную.

С: Напишите программу для перевода чисел из десятичной системы счисления в римскую и обратно.

§ 10. Создание компонентов

А: Постройте компонент для ввода целых чисел.

В: Постройте компонент для ввода целых чисел в шестнадцатеричной системе счисления.

С: Постройте компонент для ввода чисел в римской системе счисления.

§ 11. Модель и представление

А: Закончите программу из параграфа.

В: Постройте программу «Калькулятор» для выполнения вычислений с целыми числами.

С: Постройте программу «Калькулятор» для выполнения вычислений с целыми числами в римской системе счисления.

Глава 2. Программирование на языках C++ и C#

§ 12. Классы и объекты в C++

§ 13. Программа с классами (практикум)

А: Закончите программу из параграфа.

В: Доработайте программу так, чтобы снять ограничение на максимальные размеры поля.

С: Доработайте программу так, чтобы машина двигалась только влево и вправо, а дорога «прокручивалась» сверху вниз.

§ 14. Инкапсуляция

А: Закончите реализацию класса *Перо* из параграфа.

В: Измените внутреннее устройство объектов класса *Перо* так, чтобы три составляющих RGB-кода цвета хранились в виде массива из трёх элементов.

С: Измените программу из предыдущего параграфа так, чтобы все данные объектов были закрытыми.

§ 15. Наследование

А: Закончите программу из параграфа.

В: Выполните рефакторинг программы, выделив все классы в отдельный модуль.

С: Добавьте в программу из параграфа свой тип объектов.

§ 16. Полиморфизм

А: Закончите программу из параграфа.

В: Добавьте в программу из параграфа несколько хищников.

С: Измените программу так, чтобы по щелчку мыши хищник переключался в режим «блуждания» – менял направление движения случайным образом. Повторный щелчок мышью по игровому полю должен возвращать его в режим слежения за мышью.

§ 17. Взаимодействие объектов

А: Закончите программу из параграфа.

В: Добавьте в программу из параграфа взаимодействие между остальными типами объектов.

С: Вместо хищника, который следит за мышью, добавьте в модель несколько хищников, которые двигаются в случайных направлениях.

§ 18. Простая программа на C#

А: Закончите программу из параграфа.

В: Доработайте программу так, чтобы при щелчке мышью форма раскрывалась на весь экран.

С: Доработайте программу так, чтобы при изменении размеров формы постепенно изменялся её цвет.

§ 19. Использование компонентов

А: Напишите программу для перевода числа в двоичную систему счисления.

В: Напишите программу для сложения чисел в двоичной системе счисления.

С: Напишите программу для шифрования и дешифровки текста с помощью шифра простой замены.

§ 20. Создание новых классов

А: Закончите программу для вычисления арифметических выражений из параграфа.

В: Доработайте программу из параграфа так, чтобы она вычисляла значения правильных выражений со скобками.

С: Доработайте программу из параграфа так, чтобы она вычисляла значения правильных выражений со скобками и вызовами функций *sin*, *cos*, *abs*, *sqrt*.



Пронумеровано,
прошнуровано и скреплено
печатью И. Г. Исанова
_____ листа(-ов)
Директор МОБУ Лицей №6
И. Г. Исанова