Энгельсский технологический институт (филиал) федерального государственного бюджетного образовательного учреждения высшего образования «Саратовский государственный технический университет имени Гагарина Ю.А.»

Кафедра «Естественные и математические науки»

РАБОЧАЯ ПРОГРАММА

по дисциплине

Б.1.3.9.1 «Программирование в .NET»

направления подготовки
09.03.04 «Программная инженерия»
Профиль «Управление разработкой программных проектов»

форма обучения – очная
курс – 4
семестр –8
зачетных единиц – 5
часов в неделю – 4
всего часов –180 ,
в том числе:
лекции –9
практические занятия – 27
лабораторные занятия – нет
самостоятельная работа – 144
зачет с оценкой – 8 семестр
экзамен – нет
РГР – нет
курсовая работа – 8 семестр
курсовой проект – нет

Рабочая программа обсуждена на заседании кафедры
«07» июня 2021 года, протокол № 9

И.о. зав. кафедрой _____/А.С. Мостовой/

Рабочая программа утверждена на заседании УМКН
«29» июня 2021 года, протокол № 5

Председатель УМКН _____/ А.С. Мостовой /

Энгельс 2021

1. Цели и задачи дисциплины

Цель преподавания дисциплины:

Формирование у студентов теоретических знаний и практических навыков программирования на платформе .NET и языках программирования C# и F#.

Задачи изучения дисциплины:

1. Овладеть основными концепциями .NET и языка C#;
2. Иметь практические навыки в создании объектно-ориентированных приложений, приложений с пользовательским интерфейсом, сетевых и многопоточных приложений на платформе .NET;
3. Овладеть основными принципами функционального программирования на языке F#.

2. Место дисциплины в структуре ОПОП ВО

Дисциплина Б.1.3.9.1 «Программирование в .NET» представляет собой дисциплину по выбору основной профессиональной образовательной программы бакалавриата по направлению подготовки 09.03.04 «Программная инженерия».

Дисциплина читается на 4-м курсе. Зачетных единиц 5. Продолжительность курса составляет 36 аудиторных учебных часов, образованных 9 часами лекций и 27 часами практических занятий. Помимо этого, 144 часов в курсе отводится под самостоятельную работу студентов.

Требования к «входным» знаниям, умениям и навыкам обучающегося, необходимым для освоения данной дисциплины: в рамках объема школьных знаний по информатике и математике, а также знаний, полученных при изучении дисциплины программирование и дисциплины базы данных.

Основные положения дисциплины будут использованы при выполнении ВКР.

3. Требования к результатам освоения дисциплины

Изучение дисциплины Б.1.3.9.1 «Программирование в .NET» направлено на формирование у студентов следующих компетенций:

Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности (ОПК-2)

Способен разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов (ОПК-6)

В результате освоения дисциплины студент должен:

Знать:

- знать основные конструкции и идиомы языка C#, необходимые для изучения других дисциплин, предусмотренных базовым и рабочим учебными планами, а также для применения в профессиональной деятельности;
- знать основные концепции платформы .NET, такие как исполнение на виртуальной машине, сборка мусора, многопоточность в .NET, работа с графическими пользовательскими интерфейсами в .NET;

Уметь:

- уметь создавать программы, решающие задачи на языках C# и F#;

Владеть (приобрести опыт):

- навыками использования сопутствующих инструментов разработки – отладчика, библиотек модульного тестирования, инструментов разработки пользовательских интерфейсов;
- навыками в применении широкоиспользуемых технологий на платформе .NET для применения в профессиональной деятельности;

Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

Код и наименование компетенции (результат освоения)	Код и наименование индикатора достижения компетенции (составляющей компентенции)
ОПК-2. Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности	ИД-1$_{ОПК-2}$ Знает современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности.
	ИД-2$_{ОПК-2}$ Умеет выбирать современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности.
	ИД-3$_{ОПК-2}$ Имеет навыки применения современных информационных технологий и программных средств, в том числе отечественного производства, при решении задач профессиональной деятельности.
ОПК-6. Способен разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов	ИД-1$_{ОПК-6}$ Знает основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий.
	ИД-2$_{ОПК-6}$ Умеет применять языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ.
	ИД-3$_{ОПК-6}$ Имеет навыки программирования, отладки и тестирования прототипов программно-технических комплексов задач.

Код и наименование индикатора достижения компетенции	Наименование показателя оценивания (результата обучения по дисциплине)
ИД-1$_{ОПК-2}$ Знает современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности.	Формулирует алгоритм решения задачи в виде последовательности действий, выражаемой на языках F# или C#.
ИД-2$_{ОПК-2}$ Умеет выбирать современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности.	Умеет решать задачи, выбирая наиболее подходящий интерпритатор. Оценивает сложность работы алгоритма.
ИД-3$_{ОПК-2}$ Имеет навыки применения современных информационных технологий и программных средств, в том числе отечественного производства, при решении задач профессиональной деятельности.	Реализует алгоритмы решения выбранной задачи на языках F# или C#.
ИД-1$_{ОПК-6}$ Знает основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий.	Знает основные способы отладки и профилировки программ, разработанных на платформе .NET. Знание синтаксиса языков программирования F# или C#.
ИД-2$_{ОПК-6}$ Умеет применять языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ.	Проводит профилирование и отладку. Тестирует производительность и безопасность программ, разработанных на платформе .NET.
ИД-3$_{ОПК-6}$ Имеет навыки программирования, отладки и тестирования прототипов программно-технических комплексов задач.	Владение опытом создания и отладки программного приложения.

4. Распределение трудоемкости (час.) дисциплины по темам и видам занятий

№ Модуля	№ Недели	№ Темы	Наименование темы	Часы					
				Всего	Лекции	Коллоквиумы	Лабораторные	Практические	СРС
1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
8 семестр									
1	1	1	Основы языка C# и платформы .NET	15	1	-	-	4	10
1	1	2	Разработка пользовательских интерфейсов с помощью библиотек Windows Forms и WPF	23	1	-	-	4	18
1	2	3	Многопоточное программирование	34	1	-	-	4	29
2	2-3	4	Углублённое изучение платформы .NET	36	2	-	-	5	29
2	3-4	5	Веб-программирование на платформе .NET	36	2	-	-	5	29
2	4-5	6	Функциональное программирование на языке F#	36	2	-	-	5	29
Всего				180	9	-	-	27	144

5. Содержание лекционного курса

№ темы	Всего часов	№ лекции	Тема лекции. Вопросы, отрабатываемые на лекции	Учебно-методическое обеспечение
1	**2**	**3**	**4**	**5**
1	**1**	1	Введение, C#. Обзор платформы .NET. Исключения, модульное тестирование. Контейнеры и генерики. Событийно-ориентированное программирование.	**1-6**
2	**1**	1	Пользовательский интерфейс, Windows Forms. Обзор библиотеки WPF.	**1-2**
3	**1**	2	Высокоуровневое многопоточное программирование. Низкоуровневое многопоточное программирование.	**1-4**
4	**2**	2-3	Сборка мусора в .NET.	**1**
5	**2**	3-4	Обзор ASP.NET MVC Core, Entity Framework Core	**2-5**
6	**2**	4-5	F# как альтернативный язык разработки под .NET Типы и генерики в F#. Вычислительные	**6**

			выражения в F#	

6. Содержание коллоквиумов

По данной дисциплине коллоквиумы не предусмотрены.

7. Перечень практических занятий

№ темы	Всего часов	№ занятия	Тема практического занятия. Задания, вопросы, отрабатываемые на практическом занятии	Учебно-методическое обеспечение
1	**2**	**3**	**4**	**5**
1	**4**	**1-2**	Hello, world на C#. - Объявление методов, ref- и out-параметры. - Ссылочные типы и типы-значения, объявление классов и структур, разница с Java. - Модификаторы видимости классов и членов класса, наследование, интерфейсы и абстрактные классы, base, преобразования типов.	**1,3**
2	**4**	**2-3**	Изучение шаблона для WinForms-приложений в Visual Studio. Работа с библиотекой WPF	**1,3**
3	**4**	**4-5**	Изучение устройства потоков в ОС Windows. Класс Thread, пул потоков, класс Task, контекст исполнения. Отмена операций. Async/await, их использование в стандартной библиотеке. Task Parallel Library, PLINQ, потокобезопасные коллекции. Понятия гонки и взаимоблокировки. User-mode синхронизация: атомарные чтения/записи, Volatile, понятие Memory Fence. Interlocked, понятие lock-free.	**3**
4	**5**	**6-8**	Mark & Sweep-алгоритм сборки мусора, поколения в .NET, Large Object Heap, режимы и особенности настройки сборщика мусора, мониторинг. Финализаторы, SafeHandle, особенности IDisposable, внутреннее устройство финализации, ключевое слово fixed, WeakReference.	**3**
5	**5**	**8-10**	Проработка основных принципов работы веб-приложений и веб-сервисов, архитектура ASP.NET MVC Core, структура проекта, Hello, world. Язык описания правил генерации Razor. Валидация данных.	**2,3,6**
6	**5**	**11-13**	Пишем несколько несложных функций на F# вместе, в императивном и в функциональном стиле: 1. Посчитать факториал 2. Посчитать числа Фибоначчи (за линейное время) 3. Проверить, что все элемента списка различны 4. Реализовать функцию, возвращающую все элементы двоичного дерева, удовлетворяющие переданному как параметр условию 5. Описать тип «полином» и реализовать функцию, возводящую полином в заданную степень	**2,3**

8. Перечень лабораторных работ

По данной дисциплине лабораторные работы не предусмотрены.

9. Задания для самостоятельной работы студентов

№ темы	Всего Часов	Задания, вопросы, для самостоятельного изучения (задания)	Учебно-методическое обеспечение
1	**2**	**3**	**4**
1	10	Отработать объявление методов и out параметров.	1-2
2	18	С помощью Windows Forms разработать приложение с графическим интерфейсом	1-2
3	29	Проработать основные этапы многопоточного программирования.	1-2
4	29	Проработать алгоритмы сборки мусора.	1-2
5	29	Разработать веб-приложения ASP.NET Core	1-4
6	29	Пишем несколько несложных функций на F# вместе, в императивном и в функциональном стиле: 1. Посчитать факториал 2. Посчитать числа Фибоначчи (за линейное время) 3. Проверить, что все элемента списка различны 4. Реализовать функцию, возвращающую все элементы двоичного дерева, удовлетворяющие переданному как параметр условию 5. Описать тип «полином» и реализовать функцию, возводящую полином в заданную степень	1-6

10. Расчетно-графическая работа

По данной дисциплине расчетно-графическая работа не предусмотрена.

11. Курсовая работа

Примерные темы курсовых работ:

1. Программирование диалоговых приложений с использованием объектно-ориентированной библиотеки классов .NET Framework.
2. Программирование приложений, построенных на основе архитектуры "Документ/Представление"
3. Программирование приложений, поддерживающих однодокументный интерфейс (SDI).
4. Программирование приложений, поддерживающих многодокументный интерфейс (MDI).
5. Программирование графических приложений с использованием графического интерфейса устройств (GDI+).
6. Программирование многопоточных приложений.
7. Программирование с информационным сервером INTERNET (ISAPI).
8. Программирование приложений ведения баз данных с использованием библиотеки классов .NET Framework и технологии ADO.NET.
9. Программирование текстового редактора под Windows.

10. Программирование графического редактора под Windows.
11. Программирование динамически подключаемых библиотек, содержащих операции по работе со списками (односвязными, двусвязными, кольцевыми односвязными и двусвязными). Написать программу-пример использования DLL.
12. Программирование динамически подключаемых библиотек, содержащих операции по работе со структурой данных «дерево» (обход, поиск, добавление, удаление и т.д.). Написать программу-пример использования DLL.
13. Программирование сетевых приложений с использованием функций API.
14. Программирование сетевых приложений с использованием библиотеки .NET Framework.

13. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине (модулю)

В процессе освоения образовательной программы у обучающегося в ходе изучения дисциплины "Программирование в .NET" формируются компетенции ОПК-2, ОПК-6.

Уровни освоения компетенции

Индекс	Формулировка:
ОПК-2	Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности
ОПК-6	Способен разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов

Ступени уровней освоения компетенции	Отличительные признаки	Технологии формирования	Средства и технологии оценки
Пороговый (удовлетворительный)	Знает: Основы языка C# и платформы .NET. Умеет: Разрабатывать пользовательские интерфейсы с помощью библиотек Windows Forms Владеет: Навыками использования основных конструкций программирования в .NET.	Лекции, практические занятия, СРС	Практические работы выполнены в необходимом количестве и в соответствии с темами задания. В процессе выполнения заданий на экзамене студент неоднократно обращался за помощью к преподавателю. 40% положительных ответов на вопросы.

Продвинутый (хорошо)	Знает: Основы языка C# и платформы .NET, многопоточное программирование. Умеет: Разрабатывать пользовательские интерфейсы с помощью библиотек Windows Forms и WPF Владеет: Навыками использования основных конструкций программирования в .NET.	Лекции, практи-ческие занятия, СРС	Практические работы выполнены в необходимом количестве и в соответствии с темами задания. В процессе выполнения заданий на экзамене студент редко обращался за помощью к преподавателю. 70% положительных ответов на вопросы.
Высокий (отлично)	Знает: Основы языка C# и платформы .NET, многопоточное программирование, функциональное программирование на языке F# Умеет: Разрабатывать пользовательские интерфейсы с помощью библиотек Windows Forms и WPF Владеет: Навыками использования основных конструкций программирования в .NET и методами асинхронного программирования.	Лекции, практи-ческие занятия, СРС	Практические работы выполнены в необходимом количестве и в соответствии с темами задания. В процессе выполнения заданий на экзамене студент не обращался за помощью к преподавателю. 90% положительных ответов на вопросы.

Межсессионная аттестация проводится по результатам выполненных практических работ, предусмотренных учебным планом.

Рубежный контроль уровня освоения учебной дисциплины обучающимися определяется по критериям: зачтено, не зачтено.

К экзамену студенты допускаются при наличии всех практических работ, предусмотренных рабочей программой дисциплины, выполненные надлежащего качества.

Вопросы для экзамена

1. Язык C#, CLI, основы синтаксиса языка, ссылочные типы и типы-значения, преобразования типов, представление объектов в памяти
2. Методы: способы передачи параметров, абстрактные, виртуальные и статические методы, модификаторы видимости.
3. Платформа .NET: общее описание, CLR, IL, CTS

4. Сборки: понятие сборки, сильные и слабые имена, загрузка сборки, GAC, Binding redirect, MsBuild
5. NuGet, JIT, Ngen, понятие Managed Heap, AppDomain. Понятие целевой платформы, реализации: .NET Framework, Mono, .NET Core
6. Исключения: бросание, перебрасывание и обработка, библиотечные исключения, свойства исключений, хорошие практики
7. Рефлексия: загрузка сборки, создание экземпляра объекта, работа с полями и методами, dynamic
8. Модульное тестирование: популярные библиотеки, хорошие практики, mock-объекты
9. Контейнеры и генерики в .NET, энумераторы, открытые и закрытые типы, особенности статических полей в генериках, генерики и вложенные классы
10. Генерики и наследование, ограничения на параметры-типы, ковариантность и контравариантность
11. LINQ: основные методы, синтаксис, основные реализации, свои провайдеры
12. Делегаты, их внутреннее устройство, delegate chaining, Invoke, шаблонные типы делегатов из стандартной библиотеки
13. События, анонимные методы, лямбда-выражения, замыкания, каноничное объявление события, ручное управление подпиской
14. Rx.NET, интерфейсы IObservable и IObserver, холодные и горячие последовательности, Rx.NET и LINQ, Subject
15. WinForms: назначение, класс Control, обработка и валидация ввода, Data Binding, хорошие практики
16. WPF: назначение и родственные технологии. XAML: атрибуты, конвертеры типов, расширения, коллекции. Структура классов WPF, логическое и визуальное дерево.
17. WPF: зависимые свойства, routed events, команды. Data binding: конвертеры, направления привязки, валидация.
18. WPF: стили, триггеры, шаблоны, ресурсы. Геометрия контрола, задание положения контрола и преобразования системы координат
19. Потоки в .NET: классы Thread и ThreadPool, примитивы синхронизации уровня ядра: ключевое слово lock, мониторы, семафоры, WaitHandle, ManualResetEvent/AutoResetEvent, гибридные конструкции (*Slim)
20. Lock-free-программирование: основные понятия, атомарные чтения/записи, volatile, Interlocked, Compare-And-Swap
21. Класс Task, исполнение и отмена асинхронных операций. Async/await.
22. Сборка мусора, mark and sweep, поколения, Large Object Heap, когда происходит сборка мусора
23. Режимы сборки мусора: Workstation/Server, многопоточная сборка. Динамическая настройка GC, ручное управление, мониторинг
24. Финализаторы, IDisposable, using, реализация финализации, ручное управление жизнью объекта, fixed, WeakReference
25. Веб-сервисы, веб-приложения, архитектура ASP.NET MVC. Работа с БД: понятие ORM, библиотека Entity Framework
26. Язык F#: основные особенности, let-определения, кортежи, лямбды, списки, Option, взаимная рекурсия, pipe, композиция
27. Каррирование, match, виды шаблонов, последовательности, записи, размеченные объединения
28. Хвостовая рекурсия, паттерн “Аккумулятор”, Continuation Passing Style
29. Генерики в F#, автоматическое обобщение, словари операций, касты, гибкие ограничения
30. Методы отладки проблем типизации, value restriction, point-free, особенности арифметических операторов

31. ООП в F#: методы, каррирование и кортежи при передаче параметров, конструкторы, свойства, мутабельность
32. Модификаторы видимости, наследование, абстрактные классы и интерфейсы, реализация по умолчанию, объектные выражения, модули и пространства имён.

14. Образовательные технологии

На лекциях используется «проблемный» подход к изложению материала: материал каждой лекции иллюстрируется примерами, рассматриваются нестандартные ситуации, требующие решения с использованием рассматриваемого материала. При этом студенты должны активно участвовать в обсуждении вопросов, выработке решений. Для самостоятельного изучения предлагается использовать электронные ресурсы.

На практических занятиях используются следующие методы обучения и контроля усвоения материала:

- выполнение практических работ предполагает решение индивидуальных задач по дисциплине в форме практических работ, по работе оформляется отчет, описывающий процесс решения задачи в соответствии с жизненным циклом программной системы;
- каждую работу студент защищает преподавателю и получает оценку за защиту, в рамках защиты обсуждаются различные варианты решения, предложенные студентами, сравнение решений, анализ возможных ситуаций, code review.

15. ПЕРЕЧЕНЬ УЧЕБНО-МЕТОДИЧЕСКОГО ОБЕСПЕЧЕНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ

1. Казанский, А. А. Объектно-ориентированное программирование на языке Microsoft Visual C# в среде разработки Microsoft Visual Studio 2008 и .NET Framework. 4.3 : учебное пособие и практикум / А. А. Казанский. — Москва : Московский государственный строительный университет, ЭБС АСВ, 2011. — 180 с. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: https://www.iprbookshop.ru/19258.html— Режим доступа: для авторизир. пользователей

2. Разработка Windows-приложений в среде программирования Visual Studio.Net : учебно-методическое пособие по дисциплине Информатика и программирование / составители Ю. А. Воронцов, А. Г. Ерохин. — Москва : Московский технический университет связи и информатики, 2016. — 20 с. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: https://www.iprbookshop.ru/61536.html— Режим доступа: для авторизир. пользователей

3. Макаров, А. В. Common Intermediate Language и системное программирование в Microsoft.NET : учебное пособие / А. В. Макаров, С. Ю. Скоробогатов, А. М. Чеповский. — 3-е изд. — Москва, Саратов : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2020. — 397 с. — ISBN 978-5-4497-0293-7. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: https://www.iprbookshop.ru/89403.html— Режим доступа: для авторизир. пользователей

4. Столбовский, Д. Н. Разработка Web-приложений ASP.NET с использованием Visual Studio .NET : учебное пособие / Д. Н. Столбовский. — 3-е изд. — Москва, Саратов : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2020. — 375 с. — ISBN 978-5-4497-0370-5. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: https://www.iprbookshop.ru/89469.html— Режим доступа: для авторизир. пользователей.

5. Репозиторий документации по .NET, URL: https://github.com/dotnet/docs/
6. Блог “FSharp for fun and profit”, URL: https://fsharpforfunandprofit.com/

16. Материально-техническое обеспечение

1. Учебная аудитория для проведения занятий лекционного типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации.

Укомплектована специализированной мебелью и техническими средствами обучения: 20 столов, 40 стульев; рабочее место преподавателя; мультимедийная доска; проектор BENQ 631, системный блок (Atom2550/4Гб/500, клавиатура, мышь), подключенный в сеть с выходом в Интернет и доступом в информационно-образовательную среду ЭТИ (филиал) СГТУ имени Гагарина Ю.А., учебно-наглядные пособия, обеспечивающие тематические иллюстрации по рабочей программе дисциплины.

Программное обеспечение: Microsoft Windows 7, Microsoft Office 2010 (Word, Excel, PowerPoint), GoogleChrome, ПО для мультимедийной доски.

2. Учебная аудитория для проведения практических занятий, групповых и индивидуальных консультаций, текущего контроля, выполнения курсовой работы.

Укомплектована специализированной мебелью и техническими средствами обучения: 12 столов, 12 стульев; рабочее место преподавателя; маркерная доска, 12 компьютеров (I 3/ 8 Гб/ 500), мониторы 24' BENQ, LG, Philips, клавиатура, мышь). Компьютеры объединены в локальную сеть с выходом в Интернет и доступом в информационно-образовательную среду ЭТИ (филиал) СГТУ имени Гагарина Ю.А., учебно-наглядные пособия, обеспечивающие тематические иллюстрации по рабочей программе дисциплины.

Программное обеспечение: Microsoft Windows 7, Microsoft Office 2010 (Word, Excel, PowerPoint), VisualStudio; VScode, GoogleChrome.

Автор _____ к.ф.-м.н., доц. Элькин П.М.

17. Дополнения и изменения в рабочей программе

Рабочая программа пересмотрена на заседании кафедры
«____»_____ 20 ___ года, протокол № _____

Зав. кафедрой _____/_____/

Внесенные изменения утверждены на заседании УМКС/УМКН
«_____»_____ 20 __ года, протокол № ____
Председатель УМКС/УМКН _____/_____/