



**Ministerul Educației al Republicii Moldova**

**Proiectul  
“Educație de calitate în mediul rural din Moldova”**

**FORMAREA PROFESORILOR  
PENTRU IMPLEMENTAREA CURRICULUMULUI MORDERNIZAT  
DE LICEU**

**ИНФОРМАТИКА**

**Suport de curs**

**Elaborat de: Sergiu Corlat**

**СОДЕРЖАНИЕ:**

- I. Структура и функции модернизированного куррикулума ( 2010 )
- II. Формативная значимость дисциплины
- III. Методология формирования компетенций и дидактическое планирование
- IV. Корреляция компетенций: трансверсальные – специфические – субкомпетенции
- V. Дидактические стратегии преподавания – обучения - оценивания
- VI. Дидактическое планирование, ориентированное на формирование компетенций
- VII. Методические рекомендации по использованию учебников, оборудования и других существующих ресурсов (в том числе из проекта ЕСMRM) в процессе внедрения модернизированного куррикулума

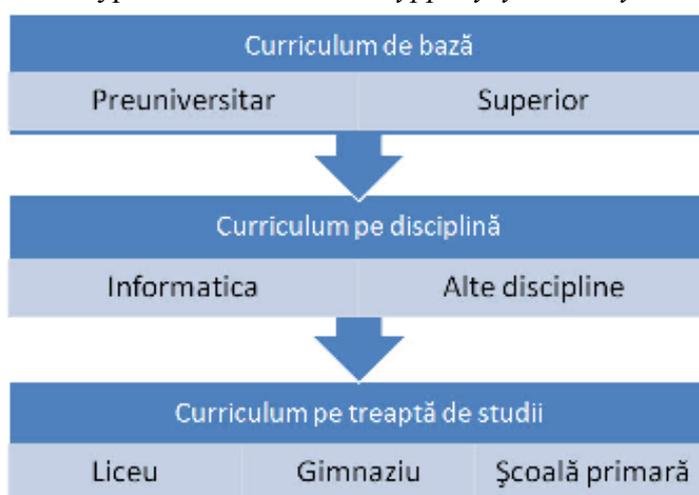
# I. СТРУКТУРА И ФУНКЦИИ МОДЕРНИЗИРОВАННОГО КУРРИКУЛУМА

**Определения.** Также как и для множества других понятий существует множество определений *куррикулума*, которые отражают различные педагогические теории и этапы развития педагогической науки. Наиболее значимыми являются:

- a) Куррикулум – вся учебная деятельность, планируемая и управляемая школой, независимо от способа (группового или индивидуального) и места (в школе или вне ее) реализации. (J.F. Kerr(Ed), *Changing the Curriculum*, University of London Press, Londra, 1968)
- b) Куррикулум – набор учебного опыта, получаемого учащимся под эгидой школы (R. Doll, *Curriculum Improvement*, Harcourt, Brace and World, New York, 1988)
- c) Куррикулум – это взаимосвязанный набор планов и опыта, реализуемых учащимся под руководством школы. (C.J. Marsh & K. Stafford, *Curriculum, Practice and Issues*, McGraw Hill, Sydney, 1988)
- d) В широком смысле, под термином *куррикулум* понимается набор образовательных процессов и учебного опыта через которые проходит учащийся за время обучения. В частном понимании *куррикулум* - это набор школьных нормативных документов, в которых указываются основные данные, относящиеся к образовательным процессам и учебному опыту, который школа предоставляет учащемуся. Этот набор обычно носит название *формального* или *официального* куррикулума (Al. Crişan, *Curriculum și dezvoltare curriculară: un posibil parcurs strategic*, в *Revista de pedagogie*, nr 3 -4/ 1994)

**Структура.** *Базовый куррикулум* описывает общие функции образовательного процесса, его задачи, цели результаты, определяет набор базовых компетенций, которые следует развивать у обучаемых. На следующем уровне находится *куррикулум по учебным дисциплинам* (курсам), который

определяет нишу учебной дисциплины в образовательном процессе, ее специфические задачи и цели в контексте получения обучаемыми специфических компетенций относящихся к данной дисциплине. *Куррикулум по уровням обучения* определяет способы развития специфических компетенций с учетом возрастных особенностей учащихся и знаний, накопленных на определенном уровне обучения (начальная школа, гимназия, лицей). (Фигура. 1)



**Функции.** Функции базового куррикулума делятся на три основные категории. Их реализация позволит всем учащимся:

- **Получить ориентированное на успех образование**, которое позволит им в последующем

- **Развивать свою личностную индивидуальность** способную к постоянному улучшению, безопасному и здоровому образу жизни, для того чтобы стать впоследствии
- **Ответственными гражданами**, которые приносят свой вклад в развитие общества.

Эти основные целеположения должны учитывать все аспекты процесса преподавания - обучения и являться отправной точкой для проектирования куррикулума.

**Ориентированное на успех образование** подразумевает воспитание учащихся, которые:

- Обладают существенными навыками обучения (язык и литература, точные и гуманитарные науки, информационные и коммуникационные технологии;
- Учатся креативно, умеют идентифицировать ресурсы, определять и решать задачи окружающего мира;
- Обладают критическим мышлением, умеют анализировать информацию, обрабатывать ее, исследовать, получать результаты и оценивать их;
- Умеют использовать различные методы общения, идентифицировать собственные ошибки и учиться на них;
- Могут обучаться самостоятельно или вместе с коллегами;
- Обладают знаниями об основополагающих для нашего мира идеях, событиях и явлениях;
- Получают удовольствие от обучения и заинтересованы в получении лучших результатов в настоящем и в будущем.

**Развитие личностной индивидуальности** подразумевает воспитание учащихся, которые:

- Обладают сознанием собственного достоинства и личности;
- Умеют общаться и формировать качественные отношения в социальной среде;
- Обладают самосознанием и могут управлять своими эмоциями;
- Обладают истинными ценностями и идеями, четкими принципами для отличия добра и зла;
- Развиваются в духе личной самостоятельности, инициативы и организованности;
- Выбирают здоровый образ жизни, развиваются физически и интеллектуально;
- Могут принимать ответственность за риски и определять условия безопасности;
- Знают свои таланты и умеют их доказывать;
- Готовы узнавать новое и максимально использовать предлагаемые возможности;
- Открыты идеям окружающего мира и человеческих достижений.

**Воспитание ответственных граждан** подразумевает воспитание учащихся, которые:

- Обладают качественной подготовкой для жизни и труда;
- Обладают предпринимательскими способностями;
- Умеют кооперировать с другими индивидуумами социальной среды;
- Уважают чужое мнение, сохраняя целостность своего собственного;
- Знакомы с собственной культурой и традициями, а также с культурой и традициями других народов, в контексте ценности человеческой цивилизации, и умеют определять нишу национальной культуры в общечеловеческой культуре;
- Могут оценить пользу многообразия;
- Не принимают выражения несправедливости, уважают права человека и принципы мирного сосуществования;
- Принимают участие в сохранении и улучшении окружающей среды на местном и глобальном уровне;

- Сообразуют свои действия в потребностями нынешних и будущих цивилизаций;
- Могут изменить мир к лучшему.

## II. ФОРМАТИВНАЯ ЗНАЧИМОСТЬ ДИСЦИПЛИНЫ ИНФОРМАТИКА

### • Определение дисциплины Информатика

Информатика участвует в формировании и развитии личности, делая упор на развитие логического и алгоритмического (структурного) мышления, формирование цифровых компетенций. Результативная деятельность в информационной среде современного общества возможна только при наличии фундаментальных знаний в области информатики и навыков инструментального использования и общения с компьютером, следствием чего является набор компетенций содержащихся в понятии информационная культура. Информатика как наука диктует необходимость общей подготовки тех, кто будет пользоваться компьютером, сетями компьютеров и информационными системами в качестве операционных инструментов, а также способствует формированию личностей, которые в своей деятельности будут управлять процессами организации деятельности в различных областях социальной деятельности.

### • Статус дисциплины в учебном плане

Информатика является обязательной дисциплиной в образовательной области „Технологии”.

### • Формативная значимость дисциплины

Целью изучения Информатики в доуниверситетском образовании является:

- формирование практических навыков работы с компьютером в целях обработки информации;
- формирование практических навыков использования компьютерных сетей и сетевых сервисов;
- формирование практических навыков общения посредством компьютерных сетей;
- изучение информатики как науки, которая способствует формированию базовых цифровых компетенций: элементы алгоритмизации, моделирования, программирования, логики, а также методов накопления, хранения и обработки информации.

### • Специфические принципы преподавания-обучения дисциплины Информатика

Лицейский курс информатики предлагает модель интегрированного изучения дисциплины, которая способствует формированию у учащихся целостной концепции об информатике как науке и методов внедрения информационных концепций для непрерывного развития современного общества.

В данном контексте, определяются следующие **специфические принципы** дисциплины Информатика:

**1. Принцип комплексного подхода к дисциплине** – структурирование учебных единиц в целостной интегрированной модульной, концентричной модели имеющей целью создание и развитие цифровых компетенций учащегося по использованию информационных систем и непрерывное формирование алгоритмического мышления.

**2. Принцип центрирования дидактической деятельности на учащемся** – принятие активной модели обучения, центрированной на учащемся, ориентированной на индивидуальную и групповую деятельность, которая способствует развитию

самостоятельности действий, оригинальности, креативности, навыков работы в команде, совмещая все с индивидуализацией ритма обучения.

**3. Принцип функциональности / социальной необходимости дидактического процесса**, который предполагает развитие навыков и компетенций необходимых для органической интеграции учащихся в информационное общество. Принцип реализуется на основе проблемных ситуаций, решение которых способствует формированию способностей к саморазвитию (самообучению).

**4. Принцип интер-дисциплинарной корреляции**, который предусматривает интердисциплинарный дидактический подход ко всем школьным дисциплинам, путем применения принципов и методов информатики для решения задач, разработки проектов, обработки специфической информации по учебным дисциплинам и использования цифровых образовательных ресурсов.

#### • **Общие направления преподавания/обучения дисциплины Информатика**

Общий процесс преподавания/обучения дисциплины Информатика разработан в контексте системы компетенций для доуниверситетского образования.

**Школьная Компетенция** – это целостная интегрированная совокупность/система знаний, умений, навыков и ценностных отношений, сформированных у учащихся в процессе обучения и используемых при необходимости, адаптированных к возрастным особенностям и когнитивному уровню учащегося для решения реальных проблем.

Менее формализованное определение определяет компетенцию как *способность решать задачи окружающего мира используя в реальном времени полученные знания и отношения.*

### **III. МЕТОДОЛОГИЯ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ**

*Формирование школьной компетенции* – длительный по времени процесс, разделяемый на условно различные пересекающиеся этапы. Традиционно эти этапы относятся к:

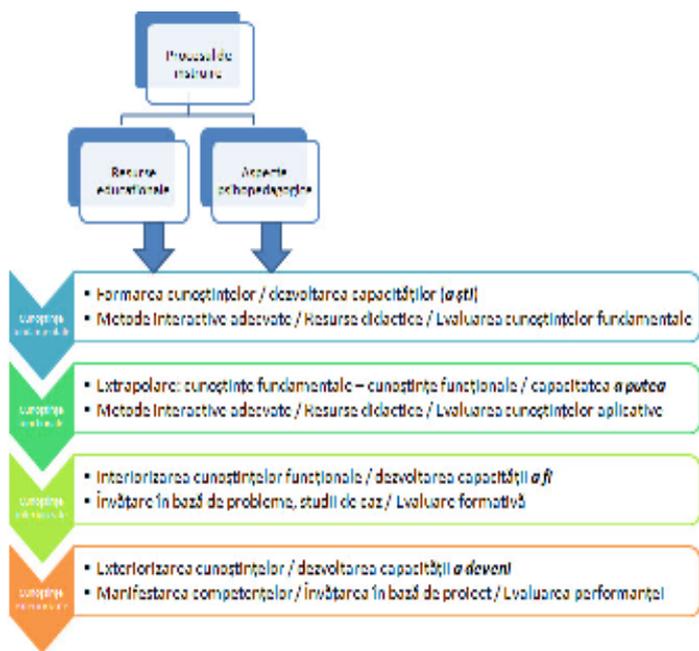
- Получению базовых знаний (знаю);
- Преобразованию знаний из базовых в функциональные (умею);
- Интернализации знаний / формированию отношений (могу быть);
- Экстернализации знаний (могу стать) – этот этап традиционно относится к научным компетенциям, которые в контексте модернизированного куррикулума становятся частью школьной деятельности.

*Этапы формирования школьной компетенции* взаимодействуют и создают непрерывный цикл, который при использовании различных методов, форм и средств обучения приводит к развитию способностей учащегося.

При проектировании куррикулума ориентированного на формирование компетенций следует учитывать:

- систему принципов специфических для научного познания с учетом возрастных особенностей;
- организацию учебного процесса на основе теорий научного познания, когнитивного развития (L. Vîgotsky), экспериментального обучения (D. Kolb);
- реализацию этапов формирования компетенций: *базовых знаний, функциональных знаний, интернализации и экстернализации знаний.*

Ранее определенные этапы и условия позволяют разработать методологическую модель формирования компетенций (специфических компетенций, субкомпетенций), фигура 2 :



**Уровень базовых знаний** основывается на восприятии, запоминании и внимании, формируя таким образом фундамент становления и развития личности. Знания основываются в первую очередь на запоминание. Информация запоминается в общем виде без вникания в суть проблемы. Учащиеся могут давать определения различных понятий, действий, законов; воспроизводить некоторые даты.

**Уровень функциональных знаний.** Позволяет обучаемому применять полученные знания. На этом уровне обучаемый использует в основном восприятие, получение опыта через органы чувств, выполняется поиск алгоритмов обработки информации, активизируются умственные процессы *анализа, синтеза, обобщения*.

**Уровень интернализации знаний.** На этом уровне развиваются умственные процессы анализа, синтеза, обобщения. Для формирования отношений широко используется моделирование, проблемные ситуации, идеализация.

**Уровень экстернализации знаний.** Это уровень на котором учащийся самостоятельно использует интернализированные знания для решения задач существенной сложности (выходящих за уровень повседневных задач), доказывая этим индивидуальные достижения.

Каждый уровень обучения, являясь одновременно и уровнем оценивания содержит три подуровня:

- элементарный** – относится к минимальному уровню знаний / умений / отношений;
- средний** – относится к наличию некоторых специальных приобретений специфических для данного уровня;
- высокий** – относится к наличию индивидуальных качеств обучаемого, необходимых для успешного прохождения уровня (этапа).

В частности, на основе ранее изложенного, для дисциплины Информатика могут быть предложены следующие методические рекомендации:

Куррикулум для лицейского образования по дисциплине Информатика проектирует и организует образовательный процесс в контексте развития специфических компетенций

относящихся к дисциплине. Такой подход предусматривает проектирование дидактического процесса для лучшего формирования у учащихся способностей к интеграции в информационное общество.

Являясь целостным набором способностей, знаний и навыков, компетенция может быть реализована только при интегральном развитии своих доминантных аспектов:

- *познавательный аспект*, который относится к использованию теорий и понятий из области Информатики;

- *функциональный аспект*, который представляет способность самовыражения личности в определенной области: профессиональной, учебной, социальной, используя цифровые средства;

- *этический аспект*, который относится к личным и социальным ценностям.

В усвоении сообщаемой информации задействованы психические процессы восприятия, запоминания и мыслительные операции. Следовательно, для разработки дидактических заданий будет использована в первую очередь таксономия Блума, ориентированная на достижение обучаемым интеллектуального уровня, заданного стандартами.

С целью усвоения учащимися знаний рекомендуется использование методов: СИНЕЛГ (SINELG), интервью, чтение предлагаемых работ, практические упражнения на компьютере, выполнение задач по развитию алгоритмического мышления;

Аспект экспертизы компетенции имеет своей ролью развитие интеллектуальных и психомоторных способностей учащихся. Для развития интеллектуального потенциала могут быть использованы таксономии: Симпсона, Дава, и др.

Методами, рекомендуемыми для дисциплины Информатика, являются: преподавание теоретического материала, работа на компьютере (индивидуально или под руководством преподавателя), решение задач, практическая работа, лабораторная работа.

Прикладной аспект компетенции формирует у учащихся отношения и поведение в контексте четко определенных социальных условий. Для достижения этой цели рекомендуется использовать таксономию Краутохла (Krauthohl).

Методами, рекомендуемыми в данном контексте являются: тематическое исследование, исследовательский проект, обсуждение и пр.

#### **IV. КОРРЕЛЯЦИЯ КОМПЕТЕНЦИЙ: ТРАНСВЕРСАЛЬНЫЕ – СПЕЦИФИЧЕСКИЕ – СУБКОМПЕТЕНЦИИ**

Ключевые / трансверсальные / междисциплинарные компетенции были определены на основе ключевых компетенций, разработанных Евро Комиссией. За некоторыми исключениями эти компетенции одинаково сформулированы во всех программах. Производным продуктом от них являются специфические компетенции, которые представляют собой детализацию ключевых компетенций для отдельной учебной дисциплины для лицейского уровня образования. Субкомпетенции – этапы формирования специфических компетенций по уровням / учебным годам.

Правильная корреляция компетенций различного уровня (трансверсальные, специфические, субкомпетенции) является основой для целостной реализации основного куррикула.

Для дисциплины Информатика, корреляция компетенций показана в Таблице 1.

В последующем будут использованы следующие нотации:

а) СТ(X) – трансверсальная компетенция с номером X;

б) CS (X) – специфическая компетенция с номером X.

Номера соответствуют индексам из следующих списков.

### **Трансверсальные компетенции:**

- СТ1. Компетенции обучения / научиться учиться.
- СТ2. Компетенции общения на родном / государственном языке.
- СТ3. Компетенции общения на иностранном языке.
- СТ4. Действенно-стратегические компетенции.
- СТ5. Компетенции самопознания и самореализации.
- СТ6. Межличностные, гражданские и нравственные компетенции.
- СТ7. Базовые компетенции по математике, наукам и технологиям.
- СТ8. Цифровые компетенции, в области технологии информации и коммуникаций (ИКТ);**
- СТ9. Общекультурные и межкультурные компетенции (восприятия и создания ценностей).
- СТ10. Предпринимательские компетенции.

### **Специфические компетенции:**

- CS1. Формирование научного восприятия компонентов информатизации современного общества.
- CS2. Знание процессов, принципов и методов кодирования и декодирования информации в целях реализации межличностной коммуникации а также коммуникационной модели человек – информатическая система.
- CS3. Идентификация общей структуры цифровых систем, принципов функционирования систем передачи, хранения и обработки информации.
- CS4. Разработка информатизированных моделей объектов, систем и процессов, часто встречаемых в повседневной деятельности.
- CS5. Применение методов алгоритмизации, формализации, анализа, синтеза и программирования для решения задач связанных с автоматизированной обработкой информации.
- CS6. Запись часто встречаемых алгоритмов в виде программ на языке программирования высокого уровня.
- CS7. Сбор, хранение и обработка информации с помощью специализированных программных приложений.
- CS8. Разработка Web документов.
- CS9. Выполнение виртуальных экспериментов, решение задач повседневной деятельности и разработка моделей изучаемых явлений, используя учебные цифровые приложения, лаборатории и обучающие средства; интерпретация полученных результатов.
- CS10. Использование компетенций в области информатики для поиска и отбора информации с целью самообразования и повышения профессиональной квалификации.
- CS11. Соблюдение авторских прав по цифровым ресурсам, этических норм и норм информационной безопасности. Защита от правонарушений в области информатики.

Элементы формирования трансверсальных компетенций СТ2, СТ3, СТ4, СТ5, СТ6, СТ9 можно идентифицировать во всех специфических компетенциях дисциплины, а также в субкомпетенциях, но они не носят доминантного характера. По этой причине в таблице корреляций включены только доминантные для реализации каждого учебного раздела куррикулума трансверсальные и специфические компетенции. Знаком (\*) отмаркированы субкомпетенции и разделы относящиеся только к реальному профилю.

**Таблица 1. Корреляция: трансверсальные / специфические / суб компетенции**

Трансверсальные компетенции	Специфические компетенции	субкомпетенции
<b>Х-й класс</b>		
<b>1. СОСТАВНЫЕ ТИПЫ ДАННЫХ</b>		
СТ1 СТ7 СТ8	CS1 CS5 CS6	<ul style="list-style-type: none"> <li>• Аргументация необходимости структурирования данных.</li> <li>• Распознавание и использование металингвистических формул и синтаксических диаграмм объявлений изучаемых типов данных.</li> <li>• Обработка структурированных данных.</li> <li>• Выбор структур данных, адекватных для решения поставленной задачи.</li> </ul>
<b>2. ИНФОРМАЦИЯ</b>		
СТ1 СТ7 СТ8	CS1 CS2 CS5 CS6	<ul style="list-style-type: none"> <li>• Идентификация базовых концептов представления и передачи информации.</li> <li>• Использование инструментов информатики в повседневной деятельности.</li> <li>• Оценка количества информации переданной, принятой и обработанной в повседневной деятельности.</li> <li>• Применение основных принципов кодирования и декодирования информации для ее передачи, получения и обработки.</li> </ul>
<b>3. АРИФМЕТИЧЕСКИЕ ОСНОВЫ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ</b>		
СТ1 СТ7 СТ8	CS1 CS2 CS5 CS6	<ul style="list-style-type: none"> <li>• Идентификация пределов представления чисел в компьютере.</li> <li>• Приложения компьютерной арифметики для решения задач обработки информации.</li> <li>• * Применение основных операций компьютерной арифметики для решения задач из области информатики.</li> </ul>
<b>4. * БУЛЕВА АЛГЕБРА</b>		
СТ1 СТ7 СТ8	CS1 CS2 CS5	<ul style="list-style-type: none"> <li>• Применение булевой алгебры для цифровой обработки информации.</li> <li>• Аргументация необходимости использования булевой алгебры в информатике.</li> </ul>
<b>5. * ЛОГИЧЕСКИЕ СХЕМЫ</b>		
СТ1 СТ7 СТ8	CS1 CS2 CS3	<ul style="list-style-type: none"> <li>• Представление логических переменных посредством физических величин.</li> <li>• Применение иерархического проектирования для разработки вычислительных структур, обработки и хранения данных</li> </ul>
<b>6. СТРУКТУРА КОМПЬЮТЕРА И КОМПЬЮТЕРНЫЕ СЕТИ</b>		

CT1 CT7 CT8 CT10	CS1 CS2 CS3 CS7 CS8 CS11	<ul style="list-style-type: none"> <li>Использование программных и аппаратных ресурсов компьютера.</li> <li>Использование внешних носителей памяти для текущего и долгосрочного хранения данных.</li> <li>Использование компьютерных устройств ввода/вывода.</li> <li>Классификация компьютерных сетей.</li> <li>Использование кооперативных технологий в сети.</li> <li>Описание принципов организации и использования сервисов Интернет.</li> </ul>
<b>XI-й класс</b>		
<b>1. ПОДПРОГРАММЫ</b>		
CT1 CT7 CT8	CS1 CS3 CS4 CS5 CS6	<ul style="list-style-type: none"> <li>Обработка данных с помощью стандартных подпрограмм и подпрограмм определенных пользователем.</li> <li>Организации передачи данных между вызывающей программой / подпрограммой и вызываемой подпрограммой.</li> <li>Структурное проектирование алгоритма и программы.</li> <li>* Использование рекурсии для решения задач.</li> </ul>
<b>2. * ДИНАМИЧЕСКИЕ СТРУКТУРЫ ДАННЫХ</b>		
CT1 CT7 CT8	CS1 CS3 CS4 CS5 CS6	<ul style="list-style-type: none"> <li>Идентификация задач, решение которых требует использования динамических структур данных.</li> <li>Использование динамических структур данных для решения задач, часто встречаемых в повседневной деятельности.</li> <li>Внедрение и обработка часто встречаемых динамических структур данных.</li> <li>Управление внутренней памятью компьютера.</li> </ul>
<b>3. * ТЕХНИКА ПРОГРАММИРОВАНИЯ</b>		
CT1 CT7 CT8	CS1 CS4 CS5 CS6	<ul style="list-style-type: none"> <li>Оценка сложности алгоритмов.</li> <li>Использование техник программирования для решения задач из различных областей.</li> <li>Выбор техники программирования, адекватной данной задаче.</li> </ul>
<b>XII-й класс</b>		
<b>1. * ЭЛЕМЕНТЫ МОДЕЛИРОВАНИЯ</b>		
CT1 CT7 CT8 CT10	CS1 CS4 CS5 CS6 CS9	<ul style="list-style-type: none"> <li>Применение критериев классификации моделей.</li> <li>Разработка математических моделей.</li> <li>Мотивация важности моделирования в экономической сфере и социальной жизни.</li> <li>Идентификация аналитических решений и решений, получаемых с помощью вычислительного эксперимента.</li> <li>Выбор типа решения в зависимости от природы задачи.</li> <li>Планирование и реализация процесса решения задач на компьютере.</li> </ul>
<b>2. * ЧИСЛЕННЫЕ МЕТОДЫ</b>		
CT1 CT7 CT8 CT10	CS1 CS4 CS5 CS6	<ul style="list-style-type: none"> <li>Идентификация точных значений и их приближений.</li> <li>Определение абсолютной и относительной погрешности.</li> <li>Оценка вычислительных погрешностей, вызванных ошибками входных данных.</li> </ul>

	CS7 CS9 CS10	<ul style="list-style-type: none"> <li>• Оценка вычислительных погрешностей, вызванных особенностями представления чисел в памяти компьютера.</li> </ul>
		<ul style="list-style-type: none"> <li>• Использование элементарных алгоритмов для разделения корней уравнения на заданном интервале.</li> <li>• Идентификация условий применения метода половинного деления (хорд, Ньютона).</li> <li>• Разработка на языке программирования высокого уровня программ итеративного вычисления корня алгебраического или трансцендентного уравнения методом половинного деления (хорд, Ньютона).</li> <li>• Выбор метода решения алгебраических и трансцендентных уравнений (половинного деления, хорд, Ньютона) наиболее подходящего для конкретной задачи</li> </ul>
		<ul style="list-style-type: none"> <li>• Выбор техники реализации алгоритма для вычисления определителей.</li> <li>• Разработка подпрограмм для численного вычисления определителей.</li> <li>• Выбор техники реализации алгоритма решения систем линейных уравнений.</li> <li>• Разработка на языке программирования высокого уровня подпрограмм для решения систем линейных уравнений.</li> </ul>
		<ul style="list-style-type: none"> <li>• Разработка программ (подпрограмм) для численного вычисления определенного интеграла методом прямоугольников</li> <li>• Идентификация задач решение которых сводится к вычислению определенного интеграла.</li> </ul>
<b>3. БАЗЫ ДАННЫХ</b>		
CT1 CT7 CT8 CT9 CT10	CS1 CS4 CS5 CS7 CS10 CS11	<ul style="list-style-type: none"> <li>• Описание структуры иерархических, сетевых и реляционных баз данных.</li> <li>• Описание структуры и функций систем управления базами данных.</li> <li>• Различение этапов разработки базы данных.</li> <li>• Знание ролей каждого из участников при разработке и использовании баз данных.</li> <li>• Проектирование структуры таблиц базы данных.</li> <li>• Создание таблиц.</li> <li>• Использование способов ввода данных в таблицах.</li> <li>• Создание и редактирование таблиц для задач из области физики, биологии, химии, географии и пр.</li> <li>• Установления корреляций между таблицами.</li> <li>• Использование операций предназначенных для сортировки записей поиска и замены значений.</li> <li>• Разработка фильтров для отбора записей.</li> </ul>
		<ul style="list-style-type: none"> <li>• Выбор адекватных типов запросов для обработки данных.</li> <li>• Создание запросов.</li> <li>• Создание запросов для отбора данных.</li> </ul>
		<ul style="list-style-type: none"> <li>• Идентификация компонент и использование правил формирования выражений.</li> <li>• Разработка запросов действия.</li> <li>• Использование техник группирования и подведения итогов..</li> </ul>

		<ul style="list-style-type: none"> <li>• Идентификация составных элементов формуляра.</li> <li>• Разработка формуляра.</li> <li>• Применение технологии модификации формуляров.</li> <li>• Использование формуляров для просмотра, редактирования и валидации данных.</li> <li>• Разработка формуляров основывающихся на коррелирующих таблицах.</li> </ul>
		<ul style="list-style-type: none"> <li>• Идентификация составных компонентов отчета.</li> <li>• Разработка отчетов и подотчетов.</li> <li>• Использование методов модификации отчетов.</li> <li>• Использование методов группировки данных в отчете.</li> </ul>
<b>4. ЭЛЕМЕНТЫ WEB ДИЗАЙНА</b>		
CT1 CT7 CT8 CT9 CT10	CS1 CS7 CS8 CS10 CS11	<ul style="list-style-type: none"> <li>• Идентификация требований и рекомендаций относительно Web документов.</li> <li>• Знание этапов разработки Web документа.</li> <li>• Идентификация Web документов в формате HTML.</li> <li>• Разработка HTML документов с помощью офисных приложений.</li> <li>• Публикация Web документов в среде Интернет.</li> </ul>
		<ul style="list-style-type: none"> <li>• Идентификация Web документов в формате HTML.</li> <li>• Разработка HTML документов с помощью офисных приложений.</li> <li>• Публикация Web документов в среде Интернет.</li> <li>• Разработка простых HTML документов.</li> <li>• Форматирование текста с помощью инструментов HTML.</li> <li>• Создание и организация списков с помощью инструментов HTML.</li> <li>• Создание и вставка внутренних и внешних ссылок (линков).</li> <li>• вставка изображений в документы HTML.</li> </ul>

## **V. ДИДАКТИЧЕСКИЕ СТРАТЕГИИ ПРЕПОДАВАНИЯ – ОБУЧЕНИЯ - ОЦЕНИВАНИЯ**

Дидактическая стратегия - обобщающий термин, который объединяет цели и виды обучения, представляя собой целостную систему средств, методов и ресурсов обучения которые могут способствовать развитию некоторых компетенций. Дидактическая стратегия занимает центральную позицию в дидактической деятельности, так как проектирование и организация урока реализуется в зависимости от стратегического решения преподавателя. Концептуально дидактическая стратегия рассматривается как комплексный дидактический сценарий, в котором присутствуют все актеры процесса преподавания – обучения, условия и методы исполнения. Следовательно стратегия устанавливает методическую траекторию для оптимального решения конкретной учебной ситуации (задачи). Таким образом дидактическая стратегия позволяет избежать ошибок, рисков и нежелательных событий в дидактической деятельности.

В качестве фактологических документов, методы являются подчиненными элементами стратегии. Другими словами, стратегия не отождествляется с методами или учебными



Остановимся на кратком описании нескольких методов, применение которых становится особенно важным в контексте внедрения куррикулума, базирующегося на компетенциях:

## 1. Проблематизированное обучение

Этот метод известен и как *обучение посредством решения задач*.

Рабочее определение метода представляет его как *дидактический метод, основанный на преднамеренном создании на пути учащегося некоторых трудностей (проблем), при преодолении которых собственными усилиями учащийся получает новые знания, навыки, отношения*.

Согласно системе классификации проблемных ситуаций по Кудрявцеву:

- 1) присутствует несоответствие между существующими знаниями учащегося и требованиями, предъявленными необходимостью решения новой задачи (ситуации);
- 2) учащемуся следует выбрать из потока или системы знаний, иногда неполных, только те которые необходимы для решения конкретной задачи (ситуации) и дополнить их недостающими, неизвестными на начальном этапе знаниями;
- 3) учащийся поставлен перед несоответствием между теоретическим методом решения задачи и сложностью практической реализации метода;
- 4) учащемуся предстоит определить динамические элементы в статической (на первый взгляд) системе;
- 5) учащемуся предстоит применить в новых для него условиях ранее приобретенные знания.

Важность использования данного метода легко обосновать исходя из множества ее свойств, органически внедряющихся в современное обучение: стимуляция формативных аспектов обучения, эффективное участие учащегося в решении задач, развитие его интереса к знаниям; развитие прикладной компоненты использования знаний учащегося; создает условия для получения навыков по трансферу знаний между областями и пр.

## 2. Учебные проекты

Учебные проекты - метод обучения / оценивания который позволяет учащимся получить существенные знания и практические навыки необходимые для успешного развития. Если мы ориентируемся на достижение главных образовательных целей XXI века, тогда этот метод должен находиться в центре внимания.

Тщательное и углубленное обучение на базе проекта организуется на базе открытого (для учащегося) вопроса или ситуации. Таким образом, учащиеся ориентируются на углубление знаний путем концентрации на важных разделах, дебатах, дискуссиях, решении задач.

Метод создает необходимость познания фундаментальных учебных содержаний и получения специфических компетенций. Обсуждение типизированных проектов (также как большинство сеансов обучения), начинается с представления знаний и концепций, а после их ассимиляции предоставляет возможность их практического применения. Реализация проекта начинается с презентации конечной цели, достижение которой требует наличие специфических знаний и навыков, создавая таким образом условия для приобретения новых знаний.

*Характеристики обучения на базе проектов:*

- 1) Учащиеся участвуют в решении комплексных задач окружающего мира; по возможности выбирают или формулируют существенные для них задачи;
- 2) Учащиеся включаются в исследовательскую деятельность, формируют навыки планирования, развивают критическое мышление и навыки решения задач - таким образом формируются компетенции реализации проектов;
- 3) Формируют специфические навыки / стандарты в контексте учебных мероприятий реализованных в проекте;
- 4) Способствует обучению и применению навыков межличностного общения в команде;
- 5) Позволяет учащимся практически использовать целый набор компетенций необходимых для дальнейшего профессионального развития и построению карьеры (планирование времени /ресурсов; личная ответственность и пр.);
- 6) Включает перспективы достижения целей обучения (относительно куррикулума) определенных в начале проекта;
- 7) Включает в себя рефлексивную деятельность обучаемых для критического анализа опыта, накопленного в проекте и его корреляции со специфическими стандартами обучения;
- 8) Заканчивается презентацией или некоторым продуктом, который доказывает выполнение поставленных учебных задач.

### 3. Тематические исследования

Рабочее определение: *метод, состоящий в изучении учащимся реальной жизненной ситуации, путем наблюдения, понимания, интерпретации которой реализуется процесс познания.*

Для классического тематического исследования идентифицируются следующие этапы:

- a) Выбор ситуации и определение основных ее элементов;
- b) Инициализация ситуации: либо в виде законченной ранее ситуации – для анализа, либо в виде открытой задачи, для которой необходимо найти готовое решение;
- c) Поиск информации существенной для данной ситуации;
- d) Систематизация материалов, используя различные методы, в том числе статистические;
- e) Дебаты, дискуссии по отобранным данным / методам / решениям;
- f) Определение выводов и собственное оценивание, презентация: выводы, гипотезы, решения.

Все описанные элементы и размерности приводят к определению метода как композитному, содержащему в себе целый набор других методов, без корреляции которых тематическое исследование не может существовать.

### **Пример: Учебный проект *сравнительный анализ эффективности алгоритмов сортировки***

Проект может быть предложен с различными уровнями сложности, в зависимости от отобранных для сравнительного анализа алгоритмов сортировки.

Для проекта **низкого уровня сложности** могут быть предложены алгоритмы *пузырьковой сортировки* и *сортировки выбором* (или *вставкой*). Проект **среднего уровня сложности** предполагает сравнение медленного алгоритма (пузырьковая сортировка, сортировка вставкой, выбором) и одного из оптимальных алгоритмов сортировки (быстрая сортировка, heap сортировка). Высокий уровень сложности проекта получается при сравнении двух оптимальных алгоритмов - *быстрая сортировка* и *heap сортировка*.

Для реализации проекта учащийся (или группа учащихся) будет нуждаться в следующих ресурсах:

- A. информационные – описания алгоритмов для выбранных методов сортировки; описания алгоритма определения времени работы фрагмента программы; алгоритм присваивания случайных значений элементам массива;
- B. Устройства / коммуникации: компьютер, доступ к сети Internet, мультимедийный проектор;
- C. Программные продукты: компилятор языка высокого уровня: FreePascal, Dev C++.

*Продолжительность проекта: 7 – 14 дней.*

*Период реализации:* после изучения темы *Анализ алгоритмов* (низкий уровень сложности) и метода *Разделяй и властвуй* (средний уровень). Для высокого уровня сложности период реализации не задается.

Этапы реализации:

- a) 0 (день 1-й): распределение проекта, объяснение понятий, терминов, требований, методов решения задач.
- b) 1 (день 8-й): ответы на вопросы, предложения, объяснения.
- c) 2 (день 15-й): презентация результатов проекта, дискуссия в классе.

**Пример (средний уровень): сравнение эффективности алгоритмов сортировки: пузырьковый метод и быстрая сортировка.**

Для реализации проекта были изучены алгоритмы *пузырьковой сортировки* (<http://en.wikipedia.org/wiki/Bubblesort>) и *быстрой сортировки - quicksort* (<http://en.wikipedia.org/wiki/Quicksort>).

Была реализована программа, которая формирует два одинаковых набора данных, которые затем сортируются независимо один от другого с помощью подпрограмм, реализующих каждый из алгоритмов:

```
{среда программирования - FreePascal}
program p001;
uses crt, dos;
const n=30000;
type t=array[1..n] of integer;
var a,b: t;
    t1,t2: real;
    i:integer;

procedure bubble; {пузырьковая сортировка}
var i,j,tmp: integer;
begin
  for i:=1 to n-1 do
    for j:=1 to n-i do
      if a[j]>a[j+1] then begin
        tmp:=a[j];
        a[j]:=a[j+1];
        a[j+1]:=tmp;
      end;
end;

procedure qsort(st,dr:integer); { быстрая сортировка }
var s,i,k,tmp : integer;
begin
  if (st < dr) then begin
    s:=b[st]; k:=st;
    for i:=st+1 to dr do
      if b[i]<s then begin
        k:=k+1;
```

```

                                tmp:=b[i];
                                b[i]:=b[k];
                                b[k]:=tmp;
                                end;

                                b[st]:=b[k];
                                b[k]:=s;
                                qsort(st,k-1);
                                qsort(k+1,dr);
                                end
                                else exit;
end;

function timpcurrent : real;
var h,m,s,ms: word;
begin
  gettime(h,m,s,ms);
  timpcurrent:=h*3600+m*60+s+ms/100;
end;

begin
  clrscr;
  {генерация данных для сортировки}
  randomize;
  for i:=1 to n do
    begin
      a[i]:= 1+random(n);
      b[i]:=a[i];
    end;
  t1:=timpcurrent; {текущее время перед вызовом процедуры сортировки}
  bubble;
  t2:=timpcurrent; { текущее время после окончания процедуры сортировки }
  writeln('metoda bulelor ', n, ' elemente. timp: ', t2-t1:0:10);
  t1:=timpcurrent;
  qsort(1,n);
  t2:=timpcurrent;
  writeln('sortarea rapida ', n, ' elemente. timp: ', t2-t1:0:10);
end.

```

Результаты:

```

Free Pascal IDE
metoda bulelor 30000 elemente. timp: 8.9100000000
sortarea rapida 30000 elemente. timp: 0.0100000000

```

**Анализ:** На одинаковом наборе из 30.000 элементов быстрая сортировка оказалась в 890 раз более эффективной чем пузырьковая. Причиной является ее средняя сложность  $O(n \log n)$  в отличие от средней сложности пузырькового метода  $O(n^2)$ .

**предложения:** в процессе презентации проекта можно предложить авторам объяснить суть каждого из использованных алгоритмов (для среднего и высокого уровня).

### Использование результатов проекта для тематического исследования:

Пусть учащийся или группа А выполнил предыдущий проект. Преподаватель может предложить учащемуся или группе В выполнить сравнительный анализ времени работы каждого из алгоритмов в зависимости от количества сортируемых элементов. Это задание

намного проще предыдущего. В качестве примера можно предложить определение необходимого времени для сортировки 3.000.000 записей о гражданах Республики Молдова, которые присутствуют в Регистре населения, используя каждый из алгоритмов отдельно.

## **VI. ДИДАКТИЧЕСКОЕ ПЛАНИРОВАНИЕ ОРИЕНТИРОВАННОЕ НА ФОРМИРОВАНИЕ КОМПЕТЕНЦИЙ**

Куррикулум по дисциплине Информатика, как составная часть национального куррикулума, является нормативным обязательным документом для реализации процесса преподавания учебной дисциплины а также рабочим инструментом преподавателя. Проектирование дидактических стратегий, которые подготавливают реализацию дидактической деятельности в классе, предполагает прохождение преподавателем следующих шагов:

### **А. При традиционном проектировании:**

- изучение куррикулума по дисциплине;
- разработка долгосрочного планирования / поурочного;
- разработка дидактических проектов индивидуальных уроков.

### **В. В контексте обучения направленного на формирование компетенций:**

- изучение куррикулума по дисциплине;
- определение учебных блоков (единиц обучения), которые ставятся в основу долгосрочного планирования;
- разработка долгосрочного планирования на основе учебных блоков;
- дидактическое проектирование учебных блоков, поурочно.

Подходы различаются структурами, которые проектируются на финальном этапе: *урок* (А) и *учебный блок* (В).

Рабочие определения этих терминов однозначно указывают на реализацию целеположений (*урок*) и формирование компетенций (*учебный блок*):

*Урок* – основная организационная форма дидактической деятельности, посредством которой учащимся передаются определенные знания в течение единицы времени; школьный час посвященный некоторой учебной дисциплине.

*Учебный блок* (единица обучения) – последовательность уроков, формирующая открытую дидактическую структуру со следующими свойствами:

- Согласуется с формированием компетенций;
- Имеет единую тематику;
- Имеет непрерывную структуру в определенном промежутке времени;
- Реализуется через заданные модели преподавания / обучения;
- Подчиняет урок как операционный элемент блока;
- Заканчивается оцениванием.

Хотя дидактическое планирование на основе единиц обучения более подходит для обучения нацеленного на формирование компетенций, в дальнейшем нужно использовать оба вида планирования, а также их вариации.

Долгосрочное планирование (на основе дидактических проектов) легко адаптируется для использования в контексте модернизации куррикулума. Основным структурным элементом традиционного планирования является столбец детализированного учебного контента (содержания), в зависимости от которого определяются реализуемые компетенции и субкомпетенции. Таким образом, традиционная модель может быть скорректирована под требования модернизированного куррикулума добавлением в структуру двух столбцов, которые будут содержать специфические компетенции и субкомпетенции, которые коррелируют с текущим учебным контентом (Таблица 2).

Методология разработки долгосрочного планирования на основе учебных блоков (единиц обучения) является относительно новой. Исходя из этого представим **алгоритм разработки долгосрочного планирования на базе учебных блоков**:

**Шаг 1: Персонализированное изучение школьного куррикулума. Свобода действий преподавателя** подразумевает выбор порядка прохождения тем / содержаний, распределения времени, учебной и оценочной деятельности.

**Шаг 2: определение учебных блоков:**

- Определяем основные темы программы;
- Определяем контент который ассоциируется с определенной темой;
- Определяем специфические компетенции / субкомпетенции которые ассоциируются с контентом данной темы;
- Детализируем учебное содержание по релевантности относительно компетенций;
- Проверяем в какой степени система компетенции – содержание позволяет организовать обоснованное оценивание, при необходимости исключаем некоторые содержания, которые будут использованы в других учебных блоках.

**Шаг 3: Примерное календарное планирование (уровень 1)**

- Составляется в начале семестра / учебного года.
- Является рамкой, которая позволяет постоянное корректирование дидактического процесса исходя из ситуации в классе. (первый уровень проектирования - Таблица 3).

**Примеры: Информатика, XII-й класс.**

**А. Адаптированное долгосрочное планирование:**

Долгосрочное планирование реализуется в соответствие с рекомендациями методического письма по дисциплине Информатика для 2010 года. Легко заметить адаптацию классической структуры, знакомой преподавателям к требованиям модернизированного куррикулума путем корреляции содержания – субкомпетенций – специфических компетенций.

**Таблица 2: Традиционное планирование по Информатике XII-й класс (один из вариантов)**

Nr.	Индексы специфических компетенций (CS) и субкомпетенций (SC)		Содержания	часы	дата	Замечания
	CS	SC				

ЭЛЕМЕНТЫ МОДЕЛИРОВАНИЯ (10 часов)					
1	CS1 CS4 CS5 CS6 CS9	<ul style="list-style-type: none"> <li>• Применение критериев классификации моделей.</li> <li>• Разработка математических моделей.</li> <li>• Мотивация важности моделирования в экономической сфере и социальной жизни.</li> <li>• Идентификация аналитических решений и решений, получаемых с помощью вычислительного эксперимента.</li> <li>• Выбор типа решения в зависимости от природы задачи.</li> <li>• Планирование и реализация процесса решения задач на компьютере.</li> </ul>	<b>Модели и моделирование.</b> Классификация моделей: материальные / идеальные / математические. Математическое моделирование	2	2 часа – оценивание
			<b>Аналитические решения и решения, получаемые с помощью вычислительного эксперимента</b>	2	
			<b>Этапы решения задач на компьютере</b> формулирование задачи / разработка математической модели / разработка алгоритма / написание программы / тестирование программы / анализ и интерпретация результатов	2	
Численные методы (24 часа)					
2	CS1 CS4 CS5 CS6 CS7 CS9 CS10	<ul style="list-style-type: none"> <li>• Идентификация точных значений и их приближений.</li> <li>• Определение абсолютной и относительной погрешности.</li> <li>• Оценка вычислительных погрешностей, вызванных ошибками входных данных.</li> <li>• Оценка вычислительных погрешностей, вызванных особенностями представления чисел в памяти компьютера.</li> </ul>	<b>Погрешности в численных методах:</b> абсолютная погрешность / относительная погрешность;	1	1 час – оценивание
			Погрешности / входных данных / округления / приближения / * метода / * задачи.	1	
			<b>Решение алгебраических и трансцендентных уравнений:</b> корень уравнения / разделение корней;	2	
		<ul style="list-style-type: none"> <li>• Использование элементарных алгоритмов для разделения корней уравнения на заданном интервале.</li> <li>• Идентификация условий применения метода половинного деления (хорд, Ньютона).</li> <li>• Разработка на языке программирования высокого уровня программ итеративного вычисления корня алгебраического или трансцендентного уравнения методом половинного деления (хорд, Ньютона).</li> <li>• Выбор метода решения алгебраических и трансцендентных уравнений (половинного деления, хорд, Ньютона) наиболее подходящего для конкретной задачи.</li> </ul>	метод половинного деления / рекуррентные формулы	2	1 час – оценивание
		метод хорд / фиксированный край;	2		
			метод Ньютона / начальное приближение.	2	
		<ul style="list-style-type: none"> <li>• Выбор техники реализации алгоритма для вычисления определителей.</li> <li>• Разработка подпрограмм для численного вычисления определителей.</li> <li>• Выбор техники реализации алгоритма решения систем</li> </ul>	<b>Вычисление определителей</b> - рекурсивные алгоритмы; - итеративные алгоритмы.	2	1 час – оценивание
		<b>Правило Крамера.</b>	2		

		<p>линейных уравнений.</p> <ul style="list-style-type: none"> <li>Разработка на языке программирования высокого уровня подпрограмм для решения систем линейных уравнений.</li> </ul>	* <b>Метод Гаусса.</b>	2		
		<ul style="list-style-type: none"> <li>Разработка программ (подпрограмм) для численного вычисления определенного интеграла методом прямоугольников</li> <li>Идентификация задач решение которых сводится к вычислению определенного интеграла.</li> </ul>	<p><b>Численное вычисление определенного интеграла.</b> Метод прямоугольников: средние прямоугольники / левые прямоугольники / правые прямоугольники.</p> <p>* Метод трапеций</p>	2		<b>1 час – оценивание</b>
<b>3. БАЗЫ ДАННЫХ</b>						
3	CS1 CS4 CS5 CS7 CS10 CS11	<ul style="list-style-type: none"> <li>Описание структуры иерархических, сетевых и реляционных баз данных.</li> <li>Описание структуры и функций систем управления базами данных.</li> <li>Различение этапов разработки базы данных.</li> </ul>	<p><b>Понятия и концепты. Типы баз данных:</b> иерархические / сетевые / реляционные.</p> <p><b>Системы управления базами данных:</b> Структура / функции</p> <p><b>Этапы разработки базы данных</b></p>	1		<b>2 часа – оценивание</b>
		<ul style="list-style-type: none"> <li>Знание ролей каждого из участников при разработке и использовании баз данных.</li> <li>Проектирование структуры таблиц базы данных.</li> <li>Создание таблиц с помощью мастера таблиц или путем самостоятельного проектирования.</li> <li>Использование способов ввода данных в таблицах.</li> <li>Создание и редактирование таблиц для задач из области физики, биологии, химии, географии и пр.</li> <li>Установления корреляций между таблицами.</li> <li>Использование операций предназначенных для сортировки записей поиска и замены значений.</li> <li>Разработка фильтров для отбора записей.</li> <li>Выбор адекватных типов запросов для обработки данных.</li> <li>Создание запросов с помощью мастера запросов или путем самостоятельного проектирования.</li> <li>Создание запросов для отбора данных.</li> <li>Идентификация компонент и использование правил формирования выражений.</li> <li>Разработка запросов действия.</li> <li>Использование техник группирования и подведения итогов.</li> <li>Идентификация составных элементов формуляра.</li> <li>Разработка формуляра с помощью мастера формуляров * или путем</li> </ul>		<p><b>Таблицы:</b> назначение и структура таблиц / создание таблиц;</p> <p>ввод данных в таблицы;</p> <p>изменение структуры таблиц / редактирование записей;</p> <p>установка корреляций между таблицами.</p> <p>сортировка записей / поиск и замена значений / создание и использование фильтров</p> <p><b>Запросы:</b> назначение и структура запросов;</p> <p>создание запросов;</p> <p>сортировка и группирование запросов.</p> <p><b>Выражения:</b> составление выражений; запросы на добавление, актуализацию и исключение / группировка и подведение итогов в запросе.</p> <p><b>Формуляры:</b> назначение и структура формуляра / создание и функционирование формуляра /</p>		

		<p>самостоятельного проектирования.</p> <ul style="list-style-type: none"> <li>• Применение технологии модификации формуляров.</li> <li>• Использование формуляров для просмотра, редактирования и валидации данных.</li> <li>• Разработка формуляров основывающихся на коррелирующих таблицах.</li> <li>• Идентификация составных компонентов отчета.</li> <li>• Разработка отчетов и подотчетов с помощью подсказки * или путем самостоятельного проектирования.</li> <li>• Использование методов модификации отчетов.</li> <li>• Использование методов группировки данных в отчете.</li> </ul>	<p>модификация формуляров / использование формуляров;</p> <p>формуляры основанные на коррелированных таблицах; форматирование данных.</p> <p><b>Отчеты:</b> назначение и структура отчетов / создание и функционирование отчета / создание отчетов на базе коррелированных таблиц / группировка и подведение итогов по данным в отчете.</p> <p>* Сжатие и восстановление базы данных. * Безопасность базы данных. * Администрирование базы данных.</p>	1		
<b>ЭЛЕМЕНТЫ WEB ДИЗАЙНА</b>						
4	CS1 CS7 CS8 CS10 CS11	<ul style="list-style-type: none"> <li>• Идентификация требований и рекомендаций относительно Web документов.</li> <li>• Знание этапов разработки Web документа.</li> <li>• Идентификация Web документов в формате HTML.</li> <li>• Разработка HTML документов с помощью офисных приложений.</li> <li>• Публикация Web документов в среде Интернет.</li> <li>• Разработка простых HTML документов.</li> <li>• Форматирование текста с помощью инструментов HTML.</li> <li>• Создание и организация списков с помощью инструментов HTML.</li> <li>• Создание и вставка внутренних и внешних ссылок (линков).</li> <li>• вставка изображений в документы HTML.</li> <li>• Создание и редактирование таблиц в документах HTML.</li> <li>• Использование таблиц для размещения на странице элементов HTML документа.</li> </ul>	<p><b>Web документы:</b> понятия и концепты / форматы Web документов / структура сайта / требования к Web документам / этапы разработки Web документов; программные приложения для создания и генерации Web документов</p> <p><b>Формат HTML.</b> Создание документов HTML с помощью офисных приложений. Публикация Web документов в среде Интернет.</p> <p><b>Общая структура HTML документа.</b> Форматирование текста: заголовки / параграфы / комментарии / физические стили / логические стили / горизонтальные линии.</p> <p><b>Списки:</b> с нумерацией / без нумерации / Определения / Типы маркеров / Типы нумерации / Вложенные списки.</p> <p><b>Ссылки</b> Внешние, внутренние: ссылка / путь / якорь / комментарии к ссылке.</p> <p><b>Изображения:</b> размеры / границы / связи / комментарии.</p> <p><b>Таблицы:</b> заголовков / строка / столбец / Ячейка / границы.</p>	1		<b>2 часа – оце- нивание</b>

## В. Долгосрочное планирование (по учебным блокам/единицам обучения):

Таблица 3. Долгосрочное планирование (по учебным блокам) по Информатике XII-й класс

№	CS	Учебный блок	содержания	часы	дата	Замечания
1	CS1 CS4 CS5 CS6 CS7 CS10	Элементы моделирования	Классификация моделей Математическое моделирование Аналитические решения и решения, получаемые с помощью вычислительного эксперимента Этапы решения задач на компьютере	10		
2	CS11	Погрешности численных методов	Погрешности численных методов	3		
3		Решение алгебраических и трансцендентных уравнений	Корни уравнения /разделение корней. Метод половинного деления. Метод хорд. Метод Ньютона;	9		
4		Определители и системы линейных уравнений	Вычисление определителей Правило Крамера. * Метод Гаусса..	7		
5		Вычисление определенного интеграла	Численное вычисление определенного интеграла. Метод прямоугольников: средние / левые / правые. * Метод трапеций	5		
6	CS1 CS7 CS8	Разработка структуры и таблиц базы данных	Понятия и концепты. Типы баз данных. Системы управления базами данных. Этапы разработки базы данных Таблицы.	8		
7	CS10	Запросы, выражения	Запросы, выражения	5		
8	CS11	Формуляры, отчеты	Формуляры, отчеты	7		
9		Web документы. Основные концепты.	Web документы. Создание документов HTML с помощью офисных приложений.	4		
10		Элементы HTML	Общая структура HTML документа Списки. Линки. Изображения. Таблицы	10		

### Дидактический проект урока (учебного блока). Второй уровень проектирования

Детализация долгосрочного планирования предполагает разработку дидактических проектов уроков или учебных блоков.

Любое действие по проектированию проходит через несколько этапов, которые соответствуют вопросам, задаваемым себе преподавателем. Схематично, элементы любого дидактического проекта описаны в прилегающей схеме:

В данном контексте, для классической модели планирования на основе уроков можно использовать и в дальнейшем существующие структурные модели дидактических проектов, например универсальную модель структурирования урока:



Этапы	Целеположения	Содержания	Дидактические технологии		Оценивание
			Задания	Учебные ситуации	
Повторение ранее приобретенных знаний					
Передача / ассимиляция новых знаний					
Фиксирование новых знаний и получение feed back -а					

Можно также использовать структуры дидактических проектов на основе моделей:

- а) *Мотивация – Изучение нового – Рефлексия – Дебрифинг (обобщение);*
- б) *Знаю – Хочу знать – Изучил;*
- с) *Прямого обучения (явного).*

Так же как и в случае долгосрочного планирования, изучим более подробно дидактическое проектирование учебного блока.

Учебный блок может содержать несколько уроков, следовательно проектирование блока должно учитывать разделение на уроки и распределение учебной деятельности внутри уроков.

Предлагаем упрощенную модель проектирования учебных блоков, с разделением дидактической деятельности по урокам, на основе следующих этапов:

*Ознакомление ⇒ Структурирование ⇒ Применение*

### 1. **Ознакомление** (Мотивация, изучение нового)

*Преподаватель:*

- *Определяет текущий уровень знаний учащихся необходимых понятий и терминов.*
- *Предлагает проблемы и проблемные ситуации, создает когнитивные конфликты.*

### 2. **Структурирование** (Объяснение, Определение сущности)

*Преподаватель:*

- *Помогает учащимся выразить свои наблюдения, сформулировать выводы.*
- *Помогает учащимся определить методы работы или развития теоретических результатов.*

### 3. **Применение** (Упражнения, Расширение)

*Преподаватель:*

- *Предлагает виды деятельности для углубления знаний по теме.*
- *Строит связи с другими учебными дисциплинами.*

Общая схема проектирования дидактической деятельности внутри учебного блока и ее корреляции с субкомпетенциями и содержаниями определяется следующим заголовком:

Дата	Единица обучения	Соответствующие субкомпетенции	Учебная деятельность	Ресурсы (материальные, процедуральные, временные)	Оценивание	Примечания

### **Пример: Учебный блок Элементы HTML (фрагмент)**

Трансверсальная компетенция: СТ8

Специфическая компетенция: CS8

Заголовок дидактического проекта будет содержать данные относительно: учебного блока (названия), соответствующих трансверсальных, специфических компетенций и субкомпетенций, преподавателе, классе, количестве уроков в учебном блоке, месте (местах) проведения, дидактических стратегий (принципы, методы, процессы)б формы организации дидактической деятельности а также другие данные, необходимые с точки зрения преподавателя. Большинство данных детализируется в дидактическом проекте, поэтому в заголовке достаточно их перечислить. Детализация деятельности с разделением на уроки описывается в таблице.

Дата	Субкомпетенции	Учебное содержание	Учебная деятельность	Ресурсы (материальные, временные), процедуральные, временные)	Оценивание	Примечания
??? Урок 1	<ul style="list-style-type: none"> <li>Разработка простых HTML документов.</li> <li>Форматирование текста с помощью инструментов HTML.</li> </ul>	<p><b>Общая структура HTML документа.</b> Форматирование текста: заголовки / параграфы / комментарии / физические стили / логические стили / горизонтальные линии</p>	<p>Организационная деятельность</p> <p><b>Ознакомление:</b> представление процесса разработки Web страниц ресурсами HTML</p> <p><b>Изучение:</b> Дискуссия по представленным ресурсам</p> <p><b>Структурирование:</b> определение разделов Web документа, идентификация тэгов для определения структуры. Создание учащимися пустого HTML документа. Просмотр и анализ результатов</p>	<p>2 min</p> <p>Видео фрагмент или простая web страница и ее код. – 5 min</p>	<p>Могут ли учащиеся определить разницу между страницей и ее кодом? Для чего нужен HTML? Какая информация является основной в HTML документах?</p> <p>Могут ли учащиеся определить структурные элементы HTML? Могут ли учащиеся создать HTML документ с заданным заголовком?</p>	
??? Урок 2			<p><b>Структурирование / Упражнения</b> определение методов разделения текста, выделение фрагментов текста и параграфов. Заполнение учащимися содержимого ранее созданного документа HTML. Просмотр и анализ</p>	<p><b>Ресурсы:</b> Слайды или схемы с описанием структуры и содержания HTML документов. Компьютер, Текстовый редактор (Notepad), Браузер (Internet Explorer) Время – 15 min</p> <p><b>Ресурсы:</b> Слайды или схемы с описанием структуры и содержания HTML документов. Компьютер, Текстовый редактор (Notepad), Браузер (Internet Explorer) Время – 20 min</p>	<p>Могут ли учащиеся вставить в HTML документ фрагменты текста и: - Разделить его на параграфы - Создавать подзаголовки - Выделять текст</p>	

		результатов			<p>- Выравнивать текст</p> <p>- Применять стили к параграфам?</p> <p>Обладают ли учащиеся компетенцией создания структуры сайта?</p> <p>Обладают ли учащиеся компетенцией отбора информации из библиографических источников и Internet?</p> <p>Могут ли учащиеся применить полученные знания для создания HTML- документов и форматирования текста?</p> <p>Могут ли учащиеся объективно оценить свои результаты и результаты коллег?</p>	
		<p><b>Расширение:</b> домашнее задание): создание страниц которые в дальнейшем составят индивидуальный проект учащегося по данному учебному блоку.</p> <p>Поиск информации для включения.</p> <p><b>Расширение:</b> разработка документов HTML и их текстового содержания. Форматирование текста.</p> <p><b>Оценивание:</b> качество форматирования текста в созданных HTML документах. (оценивание преподавателем, самооценивание, взаимооценивание)</p>	<p><b>Ресурсы:</b></p> <p>Учебник Информатика, XII-й класс</p> <p>Доступ Internet, или к библиотеке.</p> <p>Время: 3 min (класс) 1 – 2 часа (самостоятельная подготовка).</p> <p><b>Ресурсы:</b></p> <p>Компьютер, Текстовый редактор (Notepad), Браузер (Internet Explorer)</p> <p>Время – 35 min</p> <p>Время – 10 min</p>		<p>Оценивание выполненной деятельности</p> <p>...</p> <p>Оценивание выполненной деятельности</p>	
???	<ul style="list-style-type: none"> <li>Создание и организация списков с помощью инструментов HTML.</li> </ul>	<p><b>Списки:</b> с нумерацией / без нумерации / Определения / Типы маркеров / Типы нумерации / Вложенные списки.</p>	<p>Виды деятельности</p> <p>описываются также как и для предыдущего урока</p> <p>...</p>	<p>Ресурсы описываются также как и для предыдущего урока</p> <p>...</p>	<p>Оценивание выполненной деятельности</p> <p>...</p>	
???	<ul style="list-style-type: none"> <li>Создание и вставка внутренних и внешних ссылок (линков)</li> </ul>	<p><b>Ссылки</b> Внешние, внутренние: ссылка / путь / якорь / комментарии к ссылке.</p>	<p>Виды деятельности</p> <p>описываются также как и для предыдущего урока</p> <p>...</p>	<p>Ресурсы описываются также как и для предыдущего урока</p> <p>...</p>	<p>Оценивание выполненной деятельности</p> <p>...</p>	

???	<ul style="list-style-type: none"> <li>• вставка изображений в документы HTML.</li> </ul>	<p><b>Изображения:</b> размеры / границы / связи / комментарии.</p>	Виды деятельности описываются также как и для предыдущего урока	Ресурсы описываются также как и для предыдущего урока	Оценивание выполненной деятельности
???	<ul style="list-style-type: none"> <li>• Создание и редактирование таблиц в документах HTML.</li> <li>• Использование таблиц для размещения на странице элементов HTML документа.</li> </ul>	<p><b>Таблицы:</b> заголовков / строка / столбец / Ячейка / границы</p>	Виды деятельности описываются также как и для предыдущего урока	Ресурсы описываются также как и для предыдущего урока	Оценивание выполненной деятельности

Примечания:

1. Распределение времени по учебным деятельности может варьировать в зависимости от специфики класса и профиля.
2. Метод организации дидактической деятельности выбирается преподавателем в зависимости от текущей ситуации в классе и технического обеспечения учебного заведения
3. Таблица может быть дополнена преподавателем дополнительными рубриками, которые способствуют улучшению организации учебного процесса.

## VII. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО ИСПОЛЬЗОВАНИЮ УЧЕБНИКОВ

### A. Учебники и методические руководства апробированные Министерством Просвещения

#### X-й класс

##### Учебники

1. *Анатол Гремалски, Юрие Мокану, Лудмила Гремалски.* Информатика. Учебник для X класса. Штиинца, Кишинэу, 2002 sau 2007.
2. *Анатол Гремалски, Юрие Мокану, Ион Спинеи.* Информатика. Язык программирования ПАСКАЛЬ. Учебник для IX-XI классов. Штиинца, 2000.

##### Методические руководства

1. *Василе Андроник.* Информатика. Ghid de implementare a curriculum-ului modernizat în învățământul liceal. I.E.P. Știința, Chișinău, 2007

#### XI-й класс

##### Учебники

1. *Анатол Гремалски.* Информатика. Учебник для XI класса. Штиинца, Кишинэу, 2008.
2. *Анатол Гремалски.* Информатика. Методы программирования. Учебник для XI класса. Штиинца, Кишинэу, 2005.
3. *Анатол Гремалски, Юрие Мокану, Ион Спинеи.* Информатика. Язык программирования ПАСКАЛЬ. Учебник для IX-XI классов. Штиинца, 2000.

##### Методические руководства

1. *Василе Андроник.* Информатика. Ghid de implementare a curriculum-ului modernizat în învățământul liceal. I.E.P. Știința, Chișinău, 2007

#### XII-й класс

##### Учебники

Не изданы.

##### Методические руководства

1. *Василе Андроник.* Информатика. Ghid de implementare a curriculum-ului modernizat în învățământul liceal. I.E.P. Știința, Chișinău, 2007

### B. Рекомендуемые издания:

1. *Braicov A.* HTML. Ghid de inițiere. Chișinău, Editura Prut Internațional, 2008.
2. *Braicov A.* Turbo Pascal. Culegere de probleme. Chișinău, Editura Prut Internațional, 2007.
3. *Cabac V.* Elemente de modelare matematică. Chișinău, Editura Lumina, 1998.
4. *Cercez E., Șerban M.* Informatica. Iași, Editura Polirom, 2000.
5. *Cercez E., Șerban M.* Programarea în limbajul C/C++ pentru liceu. Vol I, II,III. Iași, Editura Polirom, 2006 - 2008.
6. *Corlat S., Ivanov L.* Calcul numeric. Curs de lecții. Chișinău, CCRE "Presa", 2004.
7. *Giumale C.* Introducere în analiza algoritmilor. Iași, Editura Polirom, 2004.
8. *Stiven S. Skiena, Miguel A. Revilla.* Programming Challenges. Springer Verlag, New York, 2003.

9. *Thomas Cormen, Charles Leiserson, Ronald Rivest. Introducere în algoritmi. Agora, Cluj, 2001.*
10. *Robert Sedgewick, Algorithms in C, Addison Wesley, 2002.*

### **С. Цифровые ресурсы**

1. [www.edu.md](http://www.edu.md) – официальный сайт Министерства Просвещения Республики Молдова. Содержит набор правовых, нормативных документов об образовании, регламенты и инструкции по образовательным процессам.
2. <http://stireal.edu.md> – образовательные ресурсы по точным наукам: Математика, Информатика, Физика, Химия, Биология. Предназначен для учащихся лицейских классов и преподавателей.
3. [www.solveymath.com](http://www.solveymath.com) – on-line система математических вычислений. Позволяет строить графики функций, вычислять интегралы, производные, определители; решать уравнения и системы линейных уравнений и пр. Может использоваться для решения задач по математике, физике, информатике в учебное время и при выполнении трансдисциплинарных проектов.
4. <http://campion.edu.ro> – сайт для олимпиадной подготовки учащихся в по дисциплине Информатика. Позволяет участвовать в on-line конкурсах, доступ к образовательным ресурсам в области Информатики, работу с архивом задач. Выполняет автоматическую проверку присылаемых решений. энциклопедия
5. <http://wikipedia.org> – on-line энциклопедия. Содержит статьи по разным темам, в частности по точным наукам и Информатике. Может быть доступна на различных языках, но самым полным вариантом является английский. Для автоматического перевода статей, отсутствующих в русской или румынской версиях можно использовать приложение <http://translate.google.com>.
6. <http://books.google.com> - Книжная библиотека Google. Содержит существенное количество названий из различных областей знания. Позволяет поиск нужного издания по множеству параметров, включая поиск в тексте. Найденные книги могут быть просмотрены целиком или частично, в зависимости от прав доступа полученных от системы.
7. [www.didactic.ro](http://www.didactic.ro) – сайт для преподавателей из Румынии. Содержит разнообразнейшие ресурсы для преподавателей: дидактические проекты уроков, интерактивные ресурсы, презентации и пр. Позволяет создавать форумы, группы дискуссий, организовать самообучение с помощью коллег.
8. <http://lemill.net> – портал для смешанного обучения, с встроенным репозиторием ресурсов. Позволяет поиск и использование ресурсов на портале, создание и публикацию собственных ресурсов, разработку и оценивание тестов. Содержит коммуникационный модуль. Доступен на разных языках, в том числе и на русском.