

УЧЕБНА ПРОГРАМА ПО “ПРОГРАМИРАНЕ” ЗА XII КЛАС

(ВЪВЕЖДАНЕ НА ДИСЦИПЛИНАТА ПО ПРОЕКТ “ИНОВАТИВНО УЧИЛИЩЕ”)

КРАТКО ПРЕДСТАВЯНЕ НА УЧЕБНАТА ПРОГРАМА

Обучението по **Програмиране** в гимназиален етап е насочено към овладяване на базисни знания, умения и отношения, свързани с учебния предмет, с изграждането на дигитални компетентности на ученика и с приложението им в областта на дигиталните технологии.

В този клас се придобиват систематизирани знания и умения за програмните езици Java, C#. Формират се нови знания и умения за писане на код в кодов редактор. Акцентът в обучението в XII клас е върху използването на адаптивни учебни материали, целящи да формират знания и умения за използване на програмни езици Java, C#.

Учебното съдържание е представено в следните основни теми:

- Variables and data types в Java.
- Системни методи в Java
- Условни конструкции в Java
- Масиви и цикли в Джава
- Variables and data types, Системни методи в C#
- Условни конструкции, цикли и масиви в C#

- Работа по проекти с Java: Управление на сървъри
- Работа по проекти с C#: Управление на сървъри

В програмата са включени теми за запознаване с особеностите на променливите в програмните езици, цикли, работа с условни конструкции и използване и създаване функции, работа в кодов редактор и конзола. Голяма част от времето е предвидено за разработване на проекти от учениците. Основната цел на тези теми е да представят набора от софтуерни средства, които ще бъдат изучавани и използвани в гимназиалния етап на обучение по Програмиране.

ОЧАКВАНИ РЕЗУЛТАТИ ОТ ОБУЧЕНИЕТО В КРАЯ НА КЛАСА

В края на обучението в XII клас ученикът:

- разпознава основните записвания в кода на програмни езици Java и C# , и обяснява тяхното предназначение
- демонстрира отношение на отговорен потребител при работа в Интернет среда
- реагира на съобщенията, извеждани от използваното приложение, и коригира своя код съобразно забележките
- прилага съответстващата българска терминология при описание на дейности, свързани със средата за програмиране
- описва и спазва правилата за безопасна работа с компютърна система
- изброява причините за използване на функции в програмирането
- Търси и открива причините за проблемен код: безкраен цикъл, неработещи функции, съобщения в конзолата и други.
- използва основните команди за писане на работещ и постигащ предварително зададена цел код
- Работи самостоятелно по проекти. Подготвя документацията на изработения софтуерен проект. Защишава изготвения проект.

УЧЕБНО СЪДЪРЖАНИЕ

Теми	Компетентности като очаквани резултати от обучението	Нови понятия
Тема 1: Variables and data types в Java		
1.1 Типове променливи	<ul style="list-style-type: none"> посочва основните типове на променливите в езика Конкатенира данни 	Java
1.2 Създаване на коректен код	<ul style="list-style-type: none"> Създава правилен код Работи с математическите оператори в конзолата, като спазва синтаксиса на програмния език 	
Тема 2: Системни методи в Java		
2.1 Системни методи	<ul style="list-style-type: none"> Може да обясни употребата на метод <code>'Length'</code>, и възможностите за неговото прилагане Може да обясни употребата на Методите <code>'toLowerCase'</code> & <code>'toUpperCase'</code>, и възможностите за тяхното прилагане 	Методът <code>'Length'</code> Методите <code>'toLowerCase'</code> & <code>'toUpperCase'</code>
2.2 Упражнения	<ul style="list-style-type: none"> Може да обясни употребата на Методите <code>'toLowerCase'</code> & <code>'toUpperCase'</code>, и възможностите за тяхното прилагане Пише смислен код по зададено условие, с използването на изучаваните методи 	Math.min и Math.max
Тема 3: Условни конструкции в Java		
3.1 Структура на условната конструкция	<ul style="list-style-type: none"> Може да работи с условни конструкции Открива и коригира грешки в чужд код 	
3.2 Логически оператори	<ul style="list-style-type: none"> Работи с <code>&&(и)</code>, <code> (или)</code>, <code>!(отрицание)</code> Коригира грешки в код 	
3.3 Вложени конструкции, switch	<ul style="list-style-type: none"> Работи със Switch Преработва стандартен код във вид на Ternary Conditional, и обратно Работи с вложена if конструкция Може да прецени кога е най-оптимално да употребява вложена условна 	Switch, Ternary Conditional

	конструкция	
Тема 4: Масиви и цикли в Джава		
4.1 Работа с едномерни масиви в Джава	<ul style="list-style-type: none"> • Редактира предварително зададен код с използване на масив • Разбира необходимостта от използване на масиви 	Декларация и инициализация на елементи, декларация на масив и по-късна инициализация,
4.2 Работа с двумерни масиви	<ul style="list-style-type: none"> • Обяснява разликата между едномерен и двумерен масив • Обхожда двупосочно елементите на масив, чрез използване на цикъл for • Parse командата 	едномерен масив, обхождане на масив, булева стойност, стойност на елемент, двумерен масив, обхождане на двумерен масив, Parse командата
4.3 Цикли в Джава	<ul style="list-style-type: none"> • Работи с цикъл while • Работи с цикъл do while • Работи с цикъл for 	цикъл while, цикъл do while, цикъл for в Java
Тема 5: Variables and data types, Системни методи в C#		
5.1 Задължителния код за езика	<ul style="list-style-type: none"> • Знае предназначението на задължителния код за езика • Разпознава и работи с типовете променливи • Изписва коректен код за извършване на математически операции в езика 	
5.2 Системни методи	<ul style="list-style-type: none"> • Може да обясни употребата на метод '.Length', и възможностите за неговото прилагане • Може да обясни употребата на Методите '.toLowerCase' & '.toUpperCase', и възможностите за тяхното прилагане • Може да обясни употребата на Методите '.toLowerCase' & '.toUpperCase', и възможностите за тяхното прилагане • Пише смислен код по зададено условие, с използването на изучаваните методи 	Методът '.Length' Методите '.ToLower' & '.ToUpper' Math.Min и Math.Max

Тема 6: Условни конструкции, цикли и масиви в C#		
6.1 Условни конструкции в C#	<ul style="list-style-type: none"> Работи коректно със Структурата на условната конструкция if-else Използва коректно Логическите оператори Използва Switch и вложен if 	
6.2 Цикли в C#	<ul style="list-style-type: none"> Работи с цикъл while Работи с цикъл do while Работи с цикъл for Може да прецени кой вид цикъл да използва най-оптимално, при поставено условие и ограничение 	
6.1 Масиви в C#	<ul style="list-style-type: none"> Обхождане на едномерен масив с цикъл for/foreach Обхождане на двумерен масив 	
Тема 7: Работа по проекти с Java: Управление на сървъри		
7.1 Подготовка за започване на проекта	<ul style="list-style-type: none"> Описва етапите при създаването на проекта Извършва проучване, и посочва технологиите, които са необходими за използване, за изграждане на проект по предварително зададени критерии Създава модел за решаване на заданието, поставено в проекта 	
7.2 Работа по проект, 1	<ul style="list-style-type: none"> Използва на масиви за съхраняване на потребители, сървири, команди Изработва конкретни методи за добавяне, преименуване, редактиране, листване на сървъри 	
7.3 Работа по проект, 2	<ul style="list-style-type: none"> Изработва методи за работа с потребители (вход, изтриване, добавяне) Изрисува в конзола графично схема на сървърите Изчертава в конзола таблица 	
7.4 Изготвяне на документация, защита на проект	<ul style="list-style-type: none"> Изготвя документация за софтуерния проект Презентира и защитава изготвения софтуерен проект 	Документация за софтуерен проект, защита на софтуерен проект
Тема 8: Работа по проекти със C#: Управление на сървъри		
8.1 Подготовка за започване на	<ul style="list-style-type: none"> Описва етапите при създаването на проекта Извършва проучване, и посочва технологиите, които са необходими за 	

проекта	използване, за изграждане на проект по предварително зададени критерии <ul style="list-style-type: none">• Създава модел за решаване на заданието, поставено в проекта	
8.2 Работа по проект, 1	<ul style="list-style-type: none">• Използва на масиви за съхраняване на потребители, сървири, команди• Изработва конкретни методи за добавяне, преименуване, редактиране, листване на сървъри	
8.3 Работа по проект, 2	<ul style="list-style-type: none">• Изработва методи за работа с потребители (вход, изтриване, добавяне)• Изрисува в конзола графично схема на сървърите• Изчертава в конзола таблица	
8.4 Изготвяне на документация, защита на проект	<ul style="list-style-type: none">• Изготвя документация за софтуерния проект• Презентира и защитава изготвения софтуерен проект	

ПРЕПОРЪЧИТЕЛНО ПРОЦЕНТНО РАЗПРЕДЕЛЕНИЕ НА ЗАДЪЛЖИТЕЛНИТЕ УЧЕБНИ ЧАСОВЕ ЗА ГОДИНАТА

Допълнителни уточнения за конкретния учебен предмет

Обучението се осъществява в компютърна зала, като за всеки ученик има самостоятелно работно място.

Над 50% от часовете се организират под формата на комбиниран урок, по време на който учениците изпълняват практически задачи.

Препоръчително разпределение на часовете:

За нови знания и умения	30
За упражнения в лабораторна среда	56
За обобщение	6
За контролни работи	8

СПЕЦИФИЧНИ МЕТОДИ И ФОРМИ ЗА ОЦЕНЯВАНЕ НА ПОСТИЖЕНИЯТА НА УЧЕНИЦИТЕ

Проверката и оценката на знанията и уменията в обучението по информационни технологии трябва да бъдат насочени към измерване постигането на заложените в учебната програма очаквани резултати.

Очакваните резултати от обучението са свързани с усвояването на специфична за предмета терминология, практически умения за съставяне на код в среда за визуално програмиране, и в кодов редактор, умения за аргументиране при избора на технологично средство.

Поради спецификата и разнообразния характер на очакваните резултати при оценяването на знанията и уменията на учениците могат да се използват различни методи и средства за проверка и оценка:

- Тестове, съдържащи въпроси и задачи със структуриран отговор или с ограничена свобода на отговора. Подборът на тестовите задачи трябва да се съобрази с формулираните в учебната програма очаквани резултати. Тестовете дават възможност да се обхване по-голям обем от учебното съдържание за по-кратко време. Могат да се използват за установяване на входно и изходно равнище или контролно, проведено в рамките на 20-25 минути.
- Решаване на практически задачи, решението на които се реализира на компютър в час. Този тип задачи може да съдържа отделни компоненти, които измерват усвояването на конкретни умения за работа с изучавания софтуер, умения за извличане на информация, умения за създаване на модели, умения за творческо трансформиране и представяне на различни видове информация в дигитален формат и др.
- Изпълнение на практически задачи и малки проекти за домашна работа.
- Представяне на кратко проучване на допълнителни източници по дадена тема от учебното съдържание.

- Оценяване уменията при работа по проект въз основа на зададената роля на отделния ученик при изпълнение на проекта.
- Портфолио, което може да съдържа решаваните от ученика практически задачи в часовете, домашни работи, проучвания по дадена тема, тестове. За оформянето на портфолиото учителят може да посочи кои от решаваните практически задачи ще бъдат задължително включени в него и да представи критерии за оценяване на отделните задачи и на портфолиото като цяло. Задачите, включени като задължителни компоненти, трябва да измерват постигането на формулираните в учебната програма очаквани резултати. Портфолиото може да включва и допълнителни задачи.
- Работа по практически задания и проекти

Забележка: Индивидуалното портфолио може да се използва за оценяване на отделен ученик, при условие че всеки ученик работи самостоятелно на компютър, или включва само компоненти, които ученикът разработва самостоятелно – домашни работи, проучвания, тестове.

Забележка: Предложените проекти са примерни. Преподавателят може да предложи други, които обхващат разглеждания материал.

Съотношение при формиране на срочна и годишна оценка:

Текущи оценки от устни, от писмени и от практически изпитвания върху конкретна задача	40%
Оценки от контролни (теоретични или практически) или изходно ниво	30%
Оценки от работа по проекти и индивидуално портфолио по предварително зададени критерии, домашни работи	30%

ДЕЙНОСТИ ЗА ПРИДОБИВАНЕ НА КЛЮЧОВИТЕ КОМПЕТЕНТНОСТИ, КАКТО И МЕЖДУПРЕДМЕТНИ ВРЪЗКИ

Дейности за цялата програма, които могат да се включват във всяка тема

Дейности, свързани с развитие на умения за учене:

- Поставяне на задачи за работа с фрагменти от учебните помагала или помощната информация с цел самостоятелно запознаване с елементи на изучавания материал и програмен език.
- Използване на демонстрации и експериментиране в средата на изучаваното софтуерно приложение.

Дейности, свързани с развитие на уменията за общуване на чужд език:

- Използване на английско-български и българо-английски речник за елементи от интерфейса на изучаваните софтуерни приложения.

Примерни дейности за отделни раздели и теми

Ключови компетентности	Примерни дейности и междупредметни връзки
Компетентности в областта на българския език	<ul style="list-style-type: none">● Въвеждане на текст в определените от дигиталната среда места.● Анализирание на потенциалните възможности, за решаването на конкретен проблем или проект● Създаване и записване на собствен текст - коментари - за поясняване на създадения код.
Умения за общуване на чужди езици	<ul style="list-style-type: none">● Използване на команди и код, означени както на български, така и на английски език.● Въвеждане на английските, съвместно с българските наименования, на основните елементи на изучавания приложен софтуер и интерфейс● Използване на последователност от латински букви и/или знаци за означаване на наименования на методи, означения, коментари
Математическа компетентност и	<ul style="list-style-type: none">● Използване на математически оператори за съставяне на условни конструкции, цикли и означаване на

основни компетентности в областта на природните науки и на технологиите	<ul style="list-style-type: none"> индекси; Използване на знаци за сравнение при съставяне на тялото на условна конструкция.
Дигитална компетентност	<ul style="list-style-type: none"> Обработване на информация. Разглеждане на допълнителна информация в Интернет, свързана с възможностите на програмните езици - видео материали, печатни, аудио материали. Използване дигитална идентичност. Прилагане правила за безопасна работа в дигитална среда, и защита на личния профил в Интернет пространството. Създаване на дигитално съдържание. Решаване на проблеми с използване на дигитални технологии Изучаване на логиката на дисциплината Програмиране, чиито правила са в сила за повечето програмни езици.
Умения за учене	<ul style="list-style-type: none"> Търсене и обработване на информация от различни източници. Откриване на грешки в собствен и чужд код Предлагане на повече от едно вярно решение
Социални и граждански компетентности	<ul style="list-style-type: none"> Разглеждане на информация от сайтове, свързани с безопасно използване на интернет. Разглеждане на информация от сайтове, свързани с възможностите за приложение на програмните езици
Инициативност и предприемчивост	<ul style="list-style-type: none"> Планиране, изготвяне и представяне на софтуерен проект по зададена тема.
Културна компетентност и умения за изразяване чрез творчество	<ul style="list-style-type: none"> Предлага идеи за проекти, чието решение включва изучаваните IT технологии.
Умения за подкрепа на устойчивото развитие и за здравословен начин на живот и спорт	<ul style="list-style-type: none"> Изработване на проекти - създаване на персонално портфолио, което включва изучените до момента понятия и технологии в програмирането. Предлагане на идеи за обучителни програми с елементи от познати спортове и демонстриращи здравословен начин на живот и хранене.