



ЧПГ „АК-АРКУС“ ЕООД
cognitio libertas perfectio



ecis

ETS AUTHORIZED
TOEFL iBT CENTER

Утвърждавам:.....

Петър Иванов, директор на
ЧПГ „АК-Аркус“ ЕООД

**УЧЕБНА ПРОГРАМА
ПО ИНФОРМАТИКА
ЗА IX КЛАС
ОБЩООБРАЗОВАТЕЛНА ПОДГОТОВКА,
ПРОФИЛ СОФТУЕРНИ И ХАРДУЕРНИ НАУКИ**

Настоящата учебна програма съответства напълно на изискванията на ЗПУО, на Наредба №5 от 30 ноември 2015 г. за общообразователна подготовка и на Наредба №7 от 11 август 2016 г. за профилирана подготовка. В нея са интегрирани методически препоръки от Middle Years Programme (MYP) – междинната степен на Програмата на Международния Бакалауреат. Програмата е разработка на специалисти от ЧПГ „АК-Аркус“ ЕООД, гр. Велико Търново и е част от документацията, свързана с кандидатстването на гимназията за иновативно училище през 2018 г.

Адрес:

ул. Драгоман 16, гр. Велико Търново, 5000, България, Телефон: +35962619959, Факс: +35962619961
info@ac-arcus.com; www.ac-arcus.com

УЧЕБНА ПРОГРАМА ПО ИНФОРМАТИКА ЗА IX КЛАС

ОБЩООРАЗОВАТЕЛНА ПОДГОТОВКА

Кратко представяне на учебната програма:

Информатиката е наука, която се занимава с методите за структуриране, събиране, обработка и разпространение на данни. Постиганията на тази научна област са пряко и динамично свързани със създаването и развитието на високотехнологични средства (компютри, операционни и комуникационни системи, потребителски софтуер, софтуер за разработка на приложения и др.), които са важен инструмент и/или инфраструктура, върху която функционира почти всички сфери на съвременното общество. В тази връзка учебната програма (а и обучението като цяло) по информатиката в IX клас е съобразена със система от взаимосвързани фактори, по-важните от които са:

1. Мястото на предмета в учебния план:

- брой на часовете;
- връзките и значението му (в хоризонтален и вертикален план) с другите учебни предмети, особено с математика и информационни технологии (ИТ).

2. Съвременното състояние, понятийният апарат, логическата структура, методите и средствата на научната област.

3. Възрастовите (познавателни и психологични) възможности на учениците.

4. Процесът на обучението по информатика притежава потенциални възможности за личностно израждане и развитие на ученика – формиране на абстрактно и логическо мислене, възпитание и формиране на адекватно отношение към заобикалящата действителност.

5. Учебният предмет е и възлов инструмент, с помощта на който могат да се мотивират, ориентират, привличат и развит младите хора, така че впоследствие (по-горните класове и университетите) да бъдат

създадени специалисти, способни да осигурят кадрово информационното обслужване на всички обществени сфери.

ОЧАКВАНИ РЕЗУЛТАТИ В КРАЯ НА КЛАСА:

Области на компетентности	Знания, умения и компетентности <i>В резултат на обучението си ученикът</i>
Информатика	- Познава представянето на информация във вид на статични структури от данни - Посочва примери на обекти и явления, при които е практически приложимо използването на средствата на обектно-ориентираното моделиране
Обектно-ориентирано Програмиране	- Прилага обектно-ориентиран подход при създаване на компютърна програма със средно и високо ниво на сложност. - Описва основни начини за създаване, изпълнение и тестване на програмен проект в интегрирана среда за разработка с използване на визуални графични средства - Използва библиотеките от готови компоненти - Спазва добър стил на програмиране
Графичен потребителски интерфейс	- Разбира и използва основни компоненти на среда за визуално програмиране при разработка на софтуер - Проектира графичен потребителски интерфейс с визуални средства - Избира подходяща графична компонента в съответствие с необходимата функционалност на графичния интерфейс - Умее да настройва свойствата на графичните компоненти - Програмира подразбиращи се събития за основни компоненти от графичния интерфейс
Алгоритми и структури от данни	- Разбира същността на съставните статични типове с данни - Определя подходящ тип данни за определена задача - Прилага основни управляващи конструкции - Прилага основни алгоритми за работа със стринг и едномерен масив.
Софтуерни приложения	- Използва визуално програмиране за решаване на задачи със средно и високо ниво на сложност - Създава програмни приложения с мултимедийни компоненти - Представя аргументирано разработено софтуерно приложение пред публика

УЧЕБНО СЪДЪРЖАНИЕ:

Теми	Компетентности като очаквани резултати от обучението	Нови понятия / знания
1. СИМВОЛНО ОРИЕНТИРАНИ ТИПОВЕ С ДАННИ		
1.1. Символен тип	- Познава правила за именуване на константи и променливи. - Декларира, описва и инициализира променливи и константи от символен тип. - Присвоява стойност на променлива от символен тип. - Въвежда и извежда данни от символен тип в/от текстово поле. - Използва стандартни методи работа със символи.	- ASCII кодова таблица. - код на символ - стойност на символ
1.2. Тип низ	- Прави връзка между символ и низ. - Превръща низ в съвкупност от символи и символ в низ. - Познава и използва коректно стандартните методи за работа с низ и подниз.	- дължина на низ - подниз
1.3 Едномерен масив от низ	- Декларира и инициализира масив от низ. - Въвежда и извежда стойности на масив от низ. - Реализира алгоритми за обработване на масив от низ използващи сравняване на низове, сливане на низове и др.	масив от низ
2. СЛОЖНИ АЛГОРИТМИ ЗА РАБОТА С ЕДНОМЕРЕН МАСИВ		
2.1 Преобразуване на масив	- Конструира алгоритми за промяна на масив по зададен критерий, обръщане на масив, ротация на масив наляво/надясно. - Реализира конструиранияте алгоритми със средствата, предоставени от езика за програмиране Java, като използва графичен потребителски интерфейс. - Открива и отстранява синтактични и логически грешки в програмния код.	
2.2 Работа с подобласти в масив	Конструира алгоритми за намиране на максимална/минимална площадка в масив, най-дългата нарастваща подредница и др.	

	• Реализира конструираните алгоритми със средствата, предоставени от езика за програмиране Java, като използва графичен потребителски интерфейс. • Открива и отстранява синтактични и логически грешки в програмния код.	
2.3 Създаване и използване на подмасив по зададен критерий	• Конструира на алгоритми за създаване на нов масив с различна дължина от създаден масив, по зададени критерии. • Конструира на алгоритми с използване на помощен масив. • Реализира на алгоритми със средствата, предоставени от езика за програмиране Java, като използва графичен потребителски интерфейс. • Открива и отстранява синтактични и логически грешки в програмния код.	
2.4 Търсене на елемент в масив	• Конструира алгоритъм за търсене с пълно изчерпване. • Конструира алгоритъм за търсене до първото срещане. • Реализира алгоритми за търсене със средствата, предоставени от езика за програмиране Java, като използва графичен потребителски интерфейс. • Открива и отстранява синтактични и логически грешки в програмния код. • Оценява сложността и бързината на алгоритмите за търсене.	
2.5 Сортиране на масив	• Дефинира понятието сортиран масив. • Прави разлика между възходящо и низходящо сортиране и дава примери с масиви с различни базови типове. • Използва въведените методи за сортиране на масив. • Разбира нуждата от използване на методи за сортиране, различни от въведените. • Конструира алгоритъм за сортиране на масив по метода на мехурчето. • Конструира алгоритъм за сортиране на масив по метода	• сортиран масив • възходящо сортиране • низходящо сортиране

	на селекцията. • Конструира алгоритъм за сортиране на масив чрез вмъждане. • Реализира алгоритми за търсене със средствата, предоставени от езика за програмиране Java, като използва графичен потребителски интерфейс. • Открива и отстранява синтактични и логически грешки в програмния код. • Оценява сложността и бързината на алгоритмите за сортиране. • Оценява положителните и отрицателни страни на различните алгоритми за сортиране.	
2.6 Тестване на алгоритъм с използване на масив	• Разбира значението на тестването на алгоритъм за откриване на логически грешки в програмния код или алгоритъма. • Създава проследяваща таблица (tracing table) за тестване на алгоритъм. • Дефинира тестови данни и очаквани резултати от тестването.	• проследяване на алгоритъм • проследяваща таблица
3. РАБОТА ДВА И ПОВЕЧЕ ЕДНОМЕРНИ МАСИВА		
3.1 Сливане на масиви по зададен критерий	• Конструира алгоритми за сливане на масиви по зададен критерий. • Реализира конструирания алгоритми със средствата, предоставени от езика за програмиране Java, като използва графичен потребителски интерфейс. • Открива и отстранява синтактични и логически грешки в програмния код.	• сливане на масиви
3.2 Разбиване на масив на два и повече по зададен критерий	• Конструира алгоритми за разбиване на масив на няколко подмасива • Реализира конструирания алгоритми със средствата, предоставени от езика за програмиране Java, като използва графичен потребителски интерфейс.	• разбиване на масив на подмасиви

	• Открива и отстранява синтактични и логически грешки в програмния код.	• паралелни масиви • база от данни
3.3 Използване на паралелни масиви за създаване на база от данни	• Дефинира понятието паралелни масиви. • Разбира необходимостта от използване на паралелни масиви с различни базови типове. • Дефинира понятието база от данни. • Конструира алгоритми за създаване на база от данни, използването и извеждане на резултат чрез паралелни масиви. • Реализира конструирания алгоритъм със средствата, предоставени от езика за програмиране Java, като използва графичен потребителски интерфейс. • Открива и отстранява синтактични и логически грешки в програмния код.	
4. СЪЗДАВАНЕ НА СОФТУЕРЕН ПРОЕКТ		
4.1 Анализ на проекта	• Описва етапите при реализиране на софтуерен проект. • Избира клиент за своя проект и обосновава избора си / нуждата на клиента от такава разработка. • Идентифицира и анализира нуждите на клиента • Разглежда и анализира други продукти, подобни на изборания, за да открие спецификите на това, което ще създаде. • Поставя си конкретни цели, които да бъдат постигнати с разработването на проекта. • Създава модел за решаване на проблема, поставен в заданието на проекта. • Проектира графичен потребителски интерфейс.	
4.2 Изграждане на дизайн	• Създава план с времево разпределение за създаването на разработката. • Демонстрира отлични технически умения при разработване на проекта. • Следва създадения план в точка и създава програмнен код за реализация на модела.	
4.3 Създаване на продукт (писане на программен код)		

4.4 Оценка на проекта и възможности за развитие	• Използва различни тестови техники, за да прецени дали създадения продукт работи коректно и отговаря на поставените цели. • Дава идеи за усъвършенстване на разработката си. • Изготвя документация за софтуерния проект. • Презентира и защитава готовия софтуерен проект.	
---	---	--

ПРЕПОРЪЧИТЕЛНО ПРОЦЕНТНО РАЗПРЕДЕЛЕНИЕ НА ЗАДЪЛЖИТЕЛНИТЕ УЧЕБНИ ЧАСОВЕ ЗА ГОДИНАТА:

Изисквания за провеждане на обучението:

1. Програмата предвижда обучението да се извършва на базата на език за визуално програмиране – Java. Разработена е така, че усвояването на необходимите понятия и умения да става чрез активно участие на ученика в учебния процес.
2. Препоръчва се обучението да се провежда предимно под формата на комбинирани уроци, обхващащи няколко часа със структура:
 - 2.1. Упражнения за затвърждаване на учебното съдържание, съчетани с проверка и оценка на знанията и уменията и ориентирани (при възможност) към мотивация на новото учебно съдържание.
 - 2.2. Мотивиране и въвеждане на новото учебно съдържание (теоретична и манипулативна компонента). Препоръчва се новото учебно съдържание да се дозира и въвежда от учителя като последователност от смислово обособени и логически определени части (неголеми модули). След всеки такъв модул се отделя учебно време за самостоятелна практическа работа на учениците, подпомагана от учителя.
 - 2.3. Самостоятелна работа на учениците, свързана с усвояването на теоретичната и манипулативната компонента на учебното съдържание.
 - 2.4. Анализ на хода и резултатите от урока и задаване на домашна работа.

3. Учебното време на комбинираните уроци съчетава въвеждането на нови знания с упражнения на компютър. Чисто теоретичните занятия в лекционен стил трябва да бъдат сведени до минимум. Обемът на практическата работа има превес над теоретичните части.

4. Обучението по Тема 4 предвижда учениците да разработват софтуерен проект, съдържащ проверка за коректност на входни данни, разклонени и циклични алгоритми, графичен интерфейс, форматиране на изход.

Препоръчително разпределение на часовете:

За нови знания и упражнения	30%
За обобщение, контрол и оценка	20%
Практически дейности	50%

СПЕЦИФИЧНИ МЕТОДИ И ФОРМИ ЗА ОЦЕНЯВАНЕ НА ПОСТИЖЕНИЙТА НА УЧЕНИЦИТЕ

Форми на оценяване:

- Устна

Устната форма на оценка е удачна при групови обсъждания, дискусии, генериране на идеи. Оценяват се идеите и аргументите на ученика. Не се осъществява устно изпитване, отнасящо се до запаметяване на теория.

- Писмена

Писмената форма е подходяща при проверка и оценка на теоретични знания и разбиране на синтаксиса и семантиката на даден оператор. Включват се въпроси с отворен отговор, задачи за изчисление на израз, задачи за проследяване на работата на алгоритъм или програмен фрагмент, задачи за откриване на логически грешки в алгоритъм или програмен фрагмент. Писмената форма на изпитване може да се провежда самостоятелно или в съчетание с практическа форма.

- Практическа

Тъй като предметът е с предимно практическа насоченост, това е преобладаващата форма в оценяването на постиженията на учениците. Практическо изпитване се прави на базата на портфолио, съдържащо резултатите от практическа работа в клас и разработка на проект в качеството му на изходно ниво от обучението. Оценката отразява всички етапи от практическата разработка (анализ, дизайн, програмиране, оценка и защита и др. според заданието), като за тази цел учителят подготвя съответните критериални матрици.

СЪОТНОШЕНИЕ ПРИ ФОРМИРАНЕ НА СРОЧНА И ГОДИШНА ОЦЕНКА:

Текущи оценки от работа в клас, участие в групови обсъждания и дискусии, писмено изпитване	35%
Текущи оценки от практически задания в клас	25%
Оценка на изходно ниво (проект).	40%

ДЕЙНОСТИ ЗА ПРИДОБИВАНЕ НА КЮЧОВИ КОМПЕТЕНТНОСТИ, КАКТО И МЕЖДУПРЕДМЕТНИ ВРЪЗКИ

Дейност 1.	Проучване и представяне на допълнителна информация, свързана с изучаваните теми. Учениците развиват умения за учене, социални и граждански компетентности, създават се междупредметни връзки с български език и литература.
Дейност 2.	Структуриране на статистическа информация във вид на данни в контекста на конкретна информационна задача. На учениците се дава възможност да демонстрират социални и граждански компетентности, както и компетентности в областта на българския език.
Дейност 3.	Идентифициране и дефиниране на стандартите в информатиката и използване на конвенции за именуване. Учениците демонстрират дигитални компетентности и умения за общуване на чужди езици.
Дейност 4.	Редактиране, компилиране и изпълнение на готова програма, създадена за различни сфери на обществен живот.

	Учениците развиват дигитални, математически компетентности и основни компетентности в областта на природните науки и на технологиите, социални и граждански компетентности. Осъществяват се междупредметни връзки с природни науки, информационни технологии и математика.
Дейност 5.	Редактиране на дизайн на приложение с графичен потребителски интерфейс във визуална среда за програмиране. Реализиран се междупредметни връзки на информационните технологии и изобразителното изкуство - цвят, цветови схеми, елементи на интерфейса и др.
Дейност 6.	Създаване на математически модели на ситуации, изучавани в часовете по математика и природни науки, като се използват средствата на програмирането. Учениците демонстрират математически и дигитални компетентности в областта на природните науки и на технологиите.
Дейност 7.	Описване на посочените алгоритми. Учениците развиват дигитални, математически компетентности и основни компетентности в областта на природните науки.
Дейност 8.	Въвеждане и извеждане на данни в приложения, свързани в природо-математическите дисциплини, бита и бизнеса. Учениците развиват дигитални, математически компетентности и основни компетентности в областта на природните науки и на технологиите, социални и граждански компетентности. Осъществяват се междупредметни връзки с природните науки, информационните технологии и математиката.
Дейност 9.	Реализиране на алгоритми със средствата на език за програмиране. Учениците развиват дигитални, математически компетентности и основни компетентности в областта на природните науки и на технологиите, социални и граждански компетентности.
Дейност 10.	Събиране на информация от различни източници. Трансформиране на информацията във формат на данни. Учениците развиват дигитални, математически компетентности и основни компетентности в областта на природните науки и на технологиите, социални и граждански компетентности.
Дейност 11.	Създаване на самостоятелно софтуерно приложение. Развиват се граждански компетентности - форми на поведение за успешно участие в социалния живот, разрешаване на конфликти и проблеми, умения за самостоятелно учене и събиране на информация. Разработват се общественополезни и екологични теми.