

## ИНФОРМАТИКА 8. клас

### Примерно годишно разпределение на темите от учебното съдържание

УС	Тема на урочната единица	Урочна единица НЗУ П О КО	Компетентности като очаквани резултати от обучението	Нови понятия	Контекст и дейности за всяка урочна единица	Методи и форми на оценяване по теми и/или раздели	Забелжка
<b>ОСНОВИ НА ИНФОРМАТИКА</b>							
1	1.Основи на информатиката 1.1. Информатика и компютърни науки	НЗУ – 2	Изброява и описва основните информационни дейности събиране, съхраняване, преработка и разпространение на информация и общата схема на информационните потоци. Описва понятието информационен процес и дава примери на информационни процеси, свързани с решаване на житейски задачи.  Различава понятията информация и данни. Обяснява и илюстрира с примери връзката между информация и данни. Обяснява и илюстрира с примери същността на дискретното представяне на информацията за трансформирането ѝ в данни.	– основни информационни дейности – информация, данни – дискретно представяне на информацията	Актуализиране на знания за основните информационни дейности, информация, данни.  Мотивиране и въвеждане на дискретното представяне на информацията за трансформирането ѝ в данни.	– писмено изпитване – тест (изчислителни задачи, избираем отговор и др.); – устно изпитване (описва алгоритъм с думи, посочване на очаквани грешки (синтактични, логически) проследяване на алгоритъм и др.); – практическо изпитване (задачи за пресмятане, моделиране, практическа задача).	
	1.2. Числата и техните представяния		Разширява и обобщава знанията, свързани с числата и техните представяния: – непозиционни бройни системи; – същност на позиционните бройни системи; – формат на числата в десетична, двоична и шестнадесетична бройна система. Превръща числа от десетична в двоична бройна система и обратно. Извършва събиране, изваждане и умножение на две числа в двоична бройна система.	– непозиционни бройни системи – позиционни бройни системи – двоична бройна система – шестнадесетична бройна система – експоненциален формат	Актуализиране на знания, свързани с числата и техните представяния. Въвеждане на понятията – непозиционни бройни системи; – позиционните бройни системи; – формат на числата в десетична, двоична и шестнадесетична бройна система; – експоненциален формат. Формиране на умения за превръщане на числа от десетична		
			Дава примери за използване на двоична и шестнадесетична бройна система.		в двоична бройна система и обратно. Формиране на умения за събиране, изваждане и умножение на две числа в двоична бройна система.		

	1.3. Алгоритми и езици за програмиране		<p>Дефинира понятието алгоритъм и описва основните му характеристики (резултатност, крайност, детерминираност, масовост).</p> <p>Описва и проследява несложни, линейни и разклонени алгоритми с различни средства.</p> <p>Описва същността, структурата и разновидностите на цикличните алгоритмични конструкции.</p> <p>Обяснява същността и функционалното предназначение на език за програмиране.</p> <p>Обяснява същността и предназначението на транслятор (интерпретатор, компилатор).</p> <p>Представя исторически факти, свързани със създаването и развитието на съвременните езици и среди за програмиране.</p>	<ul style="list-style-type: none"> <li>– алгоритъм</li> <li>– език за програмиране</li> <li>– транслятор</li> </ul>	<p>Въвеждане на понятието алгоритъм, основните му видове и характеристики.</p> <p>Формиране на умения за описание на алгоритми с различни средства.</p> <p>Запознаване с историята, същността и функционалното предназначение на съвременни езици за програмиране.</p>	
2	2. Основи на информатиката. Практика	П – 2	<p>Усъвършенстване на уменията за информационен процес, бройни системи, преобразуване на числата и свойства и представяне на алгоритми.</p>		<p>Затвърждаване на понятията информационен процес, бройни системи и информационни дейности. Усъвършенстване на умения за представяне на числата; преобразуване на числа от една бройна система в друга; описание на алгоритми.</p>	
3	3. Основи на информатиката. Обобщение	О – 1	<p>Обобщение на знанията за информационен процес, бройни системи, преобразуване на числата и свойства и представяне на алгоритми.</p>		<p>Обобщаване на знания и умения за информационен процес и информационни дейности; представяне на числата; бройни системи; преобразуване на числа от една бройна система в друга; описание на алгоритми.</p>	
3	4. Основи на информатиката. Контрол и оценка	КО – 1	<p>Контрол и оценка на знанията и уменията на учениците.</p>		<p>Затвърждаване на знания и умения. Оценка на знания и умения. Рефлексия.</p>	
<b>СРЕДА ЗА ВИЗУАЛНО ПРОГРАМИРАНЕ</b>						
	5. Среда за визуално програмиране	НЗУ – 2	<p>Отваря проект в интегрирана среда за визуално програмиране.</p> <p>Стартира приложение с графичен потребителски интерфейс чрез средствата на интегрирана среда за програмиране.</p>	<ul style="list-style-type: none"> <li>– интегрирана среда за програмиране</li> <li>– свързващ редактор (linker)</li> <li>– програма за откриване и</li> </ul>	<p>Въвеждане на понятието интегрирана среда за програмиране и основните ѝ компоненти.</p>	<p>– писмено изпитване – тест (открива грешки в проектиране, задава свойства на обекти от потребителски</p>

4			Разпознава основните компоненти на интегрирана среда за програмиране – графичен и текстов редактор, панел с контроли, панел за свойства на обект, панел за съобщения, панел за преглед на структурата на приложението.	отстраняване на грешки (debugger) – редактор за проектиране на дизайн на графичен потребителски интерфейс – синтактични грешки в компютърна програма – логически грешки в компютърна програма – грешки по време на изпълнение на програмата		интерфейс, описва тестови случаи; прилага и създава протоколи от оценка на потребителски интерфейс на приложение); – устно изпитване (обяснява типичната употреба на контроли в приложение, обяснява проблем, обяснява проблем, за да потърси помощ); – практическо изпитване (изпълнява практически задачи по проектиране на потребителски интерфейс, редактира потребителския интерфейс на приложение с цел подобряването му).	
5	6. Интегрирана среда за визуално програмиране. Практика	П – 2	Редактира дизайна на графичния потребителски интерфейс на приложение в интегрирана среда за програмиране. Запазва проект на приложение чрез средствата на интегрирана среда за програмиране.		Актуализиране на умения за анализ на задача с несложен математически модел. Формиране на умения за редактиране на дизайна на графичния потребителски интерфейс на приложение в интегрирана среда за разработка. Усъвършенстване на умения за определяне на подбор на компоненти и определяне на свойствата им.		
6	7. Основни етапи на създаване и изпълнение на компютърна програма. Практика	П – 2	Анализира задача с несложен математически модел. Създава математически модел за решаване на несложна задача. Съпоставя математически модел с програмно решение на даден проблем. Открива основните компоненти на математически модел в демонстрирано програмно решение на даден проблем. Стартира чрез средствата на интегрирана среда предварително подготвена		Актуализиране на умения за анализ на задача с несложен математически модел. Затвърждава понятията синтактична и логическа грешка, грешка по време на изпълнение на програмата. Формиране на умения за стартиране и тестване на готов проект.		

			<p>компютърна програма с графичен потребителски интерфейс.</p> <p>Тества предварително подготвен несложен проект.</p> <ul style="list-style-type: none"> <li>– Разпознава видовете грешки при програмиране.</li> <li>– Разчита и прави предположение за естеството на синтактична грешка в даден проект.</li> <li>– Открива и прави предположение за причината за логическа грешка в дадено приложение.</li> <li>– Открива и прави предположение за причината за грешка по време на изпълнение на приложение.</li> </ul>		<p>Затвърждаване на умения за разпознаване на грешки и причините за появяването им.</p>	
7	8. Проектиране на графичен потребителски интерфейс	НЗУ – 2	<p>Знае предназначението на основни контейнери и контроли – форма, етикет, текстово поле, бутон, диалогова кутия.</p> <p>Разпознава основни свойства на графични обекти – контроли – име, състояние, етикет, фон, настройка на шрифт и др.</p> <p>Проектира несложна форма, съдържаща етикет, текстово поле, бутон.</p> <p>Настройва основни свойства на форма, етикет, текстово поле и бутон.</p> <p>Именува обекти – контроли съгласно общоприета конвенция.</p> <p>Задава функционалност на бутон, свързана с извеждането на статично съобщение в диалогова кутия.</p>	<ul style="list-style-type: none"> <li>– графичен потребителски интерфейс</li> <li>– интерфейсен компонент (контрола)</li> <li>– контейнер на контроли</li> <li>– свойство на обект</li> <li>– метод на обект</li> </ul>	<p>Въвеждане на понятията основни елементи на потребителски интерфейс (контейнери и контроли) и основни свойства на графични обекти.</p> <p>Формиране на умения за: проектиране на несложна форма, основни настройки, именуване на обекти – контроли, задаване на функционалност на бутон.</p> <p>Оценява дизайн на потребителски интерфейс и открива грешки.</p>	
8	9. Проектиране на графичен потребителски интерфейс. Практика	П – 2	<p>Отваря проект в интегрирана среда за визуално програмиране.</p> <p>Настройва основни свойства на форма, етикет, текстово поле и бутон.</p> <p>Проектира потребителски интерфейс на приложение.</p>		<p>Усъвършенстване на умения за основни елементи на потребителски интерфейс (контейнери и контроли); основни свойства на обекти – контроли, задаване на функционалност на бутон.</p> <p>Сравнява възможности.</p> <p>Проектира самостоятелно потребителски интерфейс на приложение.</p>	

9	10. Проектиране и оценка на потребителски интерфейс. Практика	П – 2	Оценява потребителски интерфейс на приложение.		Усъвършенстване на умения при създаване, редактиране и откриване на грешки при проектиране на потребителски интерфейс и тестване на приложение за грешки.		
10	11. Проектиране на графичен потребителски интерфейс. Контрол и оценка	КО – 2	Контрол и оценка на знанията и уменията на учениците.		Затвърждаване на знания и умения. Оценяване на знания и умения. Рефлексия.		
<b>ПРОГРАМИРАНЕ</b>							
11	12. Основни типове данни	НЗУ – 2	<p><b>Тип низ (стринг)</b></p> <ul style="list-style-type: none"> <li>– Познава правила за именуване на константи и променливи.</li> <li>– Декларира, описва и инициализира променливи и константи от тип низ.</li> <li>– Присвоява стойност на променлива от тип низ.</li> <li>– Въвежда и извежда данни от тип низ в/от текстово поле.</li> <li>– Извежда данни от тип низ в/от етикет.</li> <li>– Извършва конкатенация на низове.</li> <li>– Използва стандартни методи на интерфейсни компоненти за форматиране на текст.</li> </ul> <p><b>Целочислени типове данни</b></p> <ul style="list-style-type: none"> <li>– Декларира, описва и инициализира променливи и константи от целочислен тип данни.</li> </ul>	<ul style="list-style-type: none"> <li>– символ</li> <li>– низ</li> <li>– множество на допустимите данни</li> <li>– множество на допустимите операции</li> <li>– име, тип стойност на променлива</li> <li>– име, тип и стойност на константа</li> <li>– присвояване на стойност</li> <li>– конкатенация</li> </ul> <ul style="list-style-type: none"> <li>– целочислен тип данни</li> </ul>	<p>Въвеждане на понятията низ, множество на допустими данни, множество на допустими операции, име, тип стойност на променлива/константа, присвояване на стойност, конкатенация.</p> <p>Формиране на умения за именуване, деклариране, описване, инициализиране на променливи и константи от тип низ.</p> <p>Формиране на умения за присвояване на стойност от тип низ.</p> <p>Формиране на умения за въвеждане и извеждане на данни от тип низ.</p> <p>Формиране на умения за извършване на конкатенация на низове.</p> <p>Формиране на умения за използване на стандартни методи на интерфейсни компоненти за форматиране на текст.</p> <p>Въвеждане на понятието целочислен тип данни и конвенция за именуване на константи и променливи.</p>		
			– Използва вградени функции за преобразуване на низ в цяло число и обратното.	– конвенция за именуване на константи и променливи	Формиране на умения за деклариране, описване и		

		<p>– Въвежда и извежда данни от целочислен тип.</p> <p>– Използва различни целочислени типове данни.</p> <p>– Познава целочислените аритметични операции и техния приоритет.</p> <p>– Конструира аритметични изрази, съдържащи само целочислени данни, спазвайки синтаксиса и семантиката на конкретния език за програмиране.</p> <p>– Прилага и анализира резултатите от операциите – събиране, изваждане, умножение, деление, цяла част и остатък от целочислено деление.</p> <p>– Реализира модел за решаване на задачи, базиран на целочислени типове данни.</p> <p><b>Реални типове данни</b></p> <p>– Декларира, описва и инициализира променливи и константи от реален тип.</p> <p>– Използва вградени функции за преобразуване на низ в реално число и обратното.</p> <p>– Въвежда и извежда данни от реален тип данни.</p> <p>– Използва различни реални типове данни.</p> <p>– Познава приоритетите на аритметичните операции при реални типове данни.</p> <p>– Конструира аритметични изрази, съдържащ реални типове данни, спазвайки</p>	<p>– реален тип данни</p>	<p>инициализиране на променливи и константи от целочислен тип данни.</p> <p>Формиране на умения за използване на вградени функции за преобразуване на низ в цяло число и обратното.</p> <p>Формиране на умения за въвеждане и извеждане на данни от целочислен тип.</p> <p>Актуализиране на знанията за приоритет на аритметичните операции.</p> <p>Формиране на умения за конструирание на аритметични изрази, съдържащи само целочислени данни.</p> <p>Формиране на умения за реализиране на модел за решаване на задачи с целочислен тип данни.</p> <p>Въвеждане на понятието реален тип данни за именуване на константи и променливи.</p> <p>Формиране на умения за деклариране, описване и инициализиране на променливи и константи от реален тип данни.</p> <p>Формиране на умения за използване на вградени функции за преобразуване на низ в реално число и обратното.</p> <p>Формиране на умения за въвеждане и извеждане на данни от реален</p>		
		<p>синтаксиса и семантиката на конкретния език за програмиране.</p> <p>– Прилага и анализира резултатите от операциите – събиране, изваждане, умножение, деление.</p> <p>– Реализира модел за решаване на задачи, базиран на реални типове данни.</p>		<p>тип.</p> <p>Актуализиране на знанията за приоритет на аритметичните операции с реален тип данни.</p> <p>Формиране на умения за конструирание на аритметични</p>		

			<p><b>Аритметични изрази и вградени математически функции. Приоритет на операциите.</b></p> <ul style="list-style-type: none"> <li>– Оценява числената стойност на аритметичен израз, записан на език за програмиране.</li> <li>– Записва аритметичен израз със средствата на език за програмиране.</li> <li>– Прилага и използва вградени в езика за програмиране математически функции – абсолютна стойност, повдигане на степен, закръгляване, извличане на цялата част на реално число.</li> <li>– Използва приоритет на операциите в аритметични изрази, съдържащи вградени функции.</li> <li>– Създава аритметични изрази, съдържащи различни типове данни, като се съобразява със съвместимостта им.</li> <li>– Описва синтаксис и семантика на оператор за присвояване.</li> </ul>	<ul style="list-style-type: none"> <li>– аритметичен израз в език за програмиране</li> <li>– вградени математически функции в език за програмиране</li> <li>– съвместимост на типове данни</li> <li>– форматирани изход</li> </ul>	<p>изрази, съдържащи само реални данни.</p> <p>Формиране на умения за реализиране на модел за решаване на задачи с целочислен тип данни.</p> <p>Въвеждане на понятието аритметичен израз в език за програмиране; вградени математически функции в език за програмиране, съвместимост на типове данни, форматирани изход.</p> <p>Формиране на умения за оценяване на числената стойност на аритметичен израз, записан на език за програмиране.</p> <p>Формиране на умения за запис на аритметичен израз.</p> <p>Формиране на умения за прилагане на вградени в езика за програмиране математически функции.</p> <p>Формиране на умения за създаване на аритметични изрази, съдържащи различни типове данни, като се съобразява със съвместимостта им.</p>		
					<p>Въвеждане на понятията синтаксис и семантика на оператор за присвояване.</p> <p>Формиране на умения за форматиране на изхода на реално число.</p>		

12	13. Основни типове данни. Практика	П – 2	<ul style="list-style-type: none"> <li>– Декларира, описва и инициализира променливи и константи от тип низ.</li> <li>– Присвоява стойност на променлива от тип низ.</li> <li>– Въвежда и извежда данни от тип низ в/от текстово поле.</li> <li>– Извежда данни от тип низ в/от етикет.</li> <li>– Извършва конкатенация на низове.</li> <li>– Използва стандартни методи на интерфейсни компоненти за форматиране на текст.</li> <li>– Реализира модел за решаване на задачи, базиран на реални типове данни.</li> </ul>	Компоненти на графичен потребителски интерфейс, текстови полета, променливи, типове данни, моделиране на изчисления, откриване на грешки, тестване и създаване на тестове.	Усъвършенстване на уменията за използване на елементи на потребителски интерфейс и работа със стрингове. Затвърждаване на умения за проектиране.		
13	14. Основни типове данни. Обобщение	О – 1	<ul style="list-style-type: none"> <li>– Въвежда и извежда данни от тип низ в/от текстово поле.</li> <li>– Извежда данни от тип низ в/от етикет.</li> <li>– Извършва конкатенация на низове.</li> <li>– Използва стандартни методи на интерфейсни компоненти за форматиране на текст.</li> <li>– Използва стандартни методи на интерфейсни компоненти за форматиране на текст.</li> <li>– Реализира модел за решаване на задачи, базиран на целочислени и реални типове данни.</li> <li>– Форматира изхода на цяло, реално число.</li> </ul>		Усъвършенстване на умения за типове данни: тип низ (стринг), целочислени типове, реални типове, аритметични изрази и вградени математически функции. Приоритет на операциите.		
13	15. Основни типове данни. Контрол и оценка	КО – 1	Контрол и оценка на знанията и уменията на учениците.		Контрол и оценка на знанията и уменията за типове данни.		
14	16. Създаване на компютърна програма за решаване на конкретна задача	НЗУ – 2	<ul style="list-style-type: none"> <li>– Знае основните етапи при създаване на компютърна програма.</li> <li>– Анализира и проектира решението на конкретна задача.</li> <li>– Създава математическия модел за решаване на задачата.</li> </ul>	<ul style="list-style-type: none"> <li>– коментари</li> <li>– оформяне на програмния код</li> </ul>	Въвеждане на етапите за създаване на компютърна програма. Формиране на умения за анализиране и проектиране на решението на задача.		
			<ul style="list-style-type: none"> <li>– Разработва алгоритъм за решаване на задачата.</li> <li>– Определя входно – изходни данни и техните типове.</li> <li>– Структурира и разработва графичен интерфейс, като използва обекти и декларира променливи.</li> </ul>		<p>Усъвършенстване на умения за създаване на математически модел на задача.</p> <p>Формиране на умения за разработка на алгоритъм за решаване на задача.</p> <p>Въвеждане на понятията</p>		



			<ul style="list-style-type: none"> <li>– Създава и описва програмния код;</li> <li>– коментари;</li> <li>– оформяне на програмния код.</li> <li>– Стартира, тества и валидира готовия проект.</li> <li>– Открива синтактични и логически грешки в програмата.</li> <li>– Отстранява синтактични и логически грешки при програмиране.</li> <li>– Спазва изисквания за оформяне на програмния код, включващи подравняване, коментари, именуване на програмните единици.</li> </ul>		<p>входно-изходни данни и типовете им.</p> <p>Формиране на умения за разработване на графичен интерфейс с деклариране на променливи.</p> <p>Формиране на умения за стартиране, тестване и валидиране на готов проект.</p> <p>Формиране на умения за откриване на грешки при програмиране.</p> <p>Формиране на умения за спазване на изискванията за оформяне на програмен код.</p>		
15	17. Създаване на компютърна програма за решаване на конкретна задача. Практика	П – 2	<ul style="list-style-type: none"> <li>– Знае основните етапи при създаване на компютърна програма.</li> <li>– Анализира и проектира решението на конкретна задача.</li> <li>– Създава математическия модел за решаване на задачата.</li> <li>– Разработва алгоритъм за решаване на задачата.</li> <li>– Определя входно-изходни данни и техните типове.</li> <li>– Структурира и разработва графичен интерфейс, като използва обекти и декларира променливи.</li> <li>– Създава и описва програмния код;</li> <li>– коментари;</li> <li>– оформяне на програмния код.</li> <li>– Стартира, тества и валидира готовия проект.</li> <li>– Открива синтактични и логически грешки в програмата.</li> <li>– Отстранява синтактични и логически грешки при програмиране.</li> </ul>		<p>Актуализира знания за основните етапи при създаване на компютърна програма, анализ на проблема и проектиране на решението.</p> <p>Усъвършенстване на умения за създаване на компютърна програма за решаване на конкретна задача.</p>		
			<ul style="list-style-type: none"> <li>– Спазва изисквания за оформяне на програмния код, включващи подравняване, коментари, именуване на програмните единици.</li> </ul>				
	18. Програмни конструкции за реализация на разклонен алгоритъм	НЗУ – 2	<p><b>Булев тип данни</b></p> <ul style="list-style-type: none"> <li>– Обосновава необходимостта от разклоняване на алгоритмичния процес.</li> <li>– Използва константите от булев тип данни.</li> <li>– Дава примери, в които се използва булев тип данни.</li> <li>– Декларира булева променлива.</li> </ul>	<ul style="list-style-type: none"> <li>– булев тип данни</li> <li>– булеви константи – false, true</li> <li>– логически операции</li> <li>– приоритет на логическите операции</li> <li>– булев израз</li> </ul>	<p>Въвеждане на понятието булев тип данни, булеви константи, логически операции, приоритет, булев израз.</p> <p>Формиране на умения за използване на константи и променливи от булев тип.</p>		

16		<p>– Присвоява стойност на булева променлива.</p> <p>– Записва на език за програмиране булев израз, съдържащ операция за сравнение.</p> <p>– Изписва синтактично правилно на езика за програмиране основните логически операции – логическо отрицание, дизюнкция, конюнкция.</p> <p>– Познава приоритета на логическите операции.</p> <p>– Пресмята без използване на компютър стойността на булев израз.</p> <p>– Съставя със средствата на език за програмиране сложен булев израз, отговарящ на дадена логическа ситуация.</p> <p><b>Условен оператор</b></p> <p>Описва синтаксиса и семантиката на кратка и пълна форма на условен оператор. Описва разклонен алгоритъм с помощта на условен оператор.</p> <p>Използва условен оператор за проверка на коректността на входните данни за програма.</p> <p>Използва условен оператор за обработка на свойства на радиобутон и поле за отметка.</p> <p><b>Вложени условни оператори</b></p>	<p>– условен оператор</p> <p>– съставен оператор</p>	<p>Формиране на умения за запис на булев израз.</p> <p>Формиране на умения за пресмятане на стойности на булев израз.</p> <p>Формиране на умения за съставяне на програма със сложен булев израз, отговарящ на логическа ситуация.</p> <p>Въвеждане на понятието условен оператор и съставен оператор.</p> <p>Формиране на умения за използване на условен оператор за проверка на коректността на входните данни.</p> <p>Формиране на умения за използване на условен оператор за обработка на радиобутон и поле за отметка.</p> <p>Въвеждане на понятието вложен условен оператор.</p>		
----	--	---	--	--	--	--

			<ul style="list-style-type: none"> <li>– Обяснява семантиката на вложени условни оператори в кратка и пълна форма.</li> <li>– Проиграва изпълнението на фрагмент на програма, съдържаща вложени условни оператори.</li> <li>– Записва синтактично и логически правилно вложени условни оператори.</li> <li>– Създава модел и алгоритъм за решаване на задача чрез използване на вложени условни конструкции.</li> <li>– Реализира модел за решаване на задача чрез използване на вложени условни оператори.</li> <li>– Заменя вложен условен оператор с единичен и обратно.</li> </ul>	– вложен условен оператор	<p>Формиране на умения за проиграване на фрагмент от програма, съдържаща вложени условни оператори.</p> <p>Формиране на умения за създаване на модел и алгоритъм за решаване на задача с използване на вложен условен оператор.</p> <p>Формиране на умения за реализиране на модел за решаване на задача с вложен условен оператор.</p> <p>Формиране на умения за замяна на вложен условен оператор с единичен и обратно.</p>		
17	19. Програмни конструкции за реализация на разклонен алгоритъм. Практика	П – 2	<ul style="list-style-type: none"> <li>– Записва на език за програмиране булев израз, съдържащ операция за сравнение.</li> <li>– Пресмята без използване на компютър стойността на булев израз.</li> <li>– Съставя със средствата на език за програмиране сложен булев израз, отговарящ на дадена логическа ситуация.</li> <li>– Описва синтаксиса и семантиката на кратка и пълна форма на условен оператор.</li> <li>– Описва разклонен алгоритъм с помощта на условен оператор.</li> </ul>		Усъвършенстване на умения за реализиране на модел за решаване на задача с вложен условен оператор.		
18	20. Програмни конструкции за реализация на разклонен алгоритъм. Практика	П – 2	<ul style="list-style-type: none"> <li>– Използва условен оператор за проверка на коректността на входните данни за програма.</li> <li>– Използва условен оператор за обработка на свойства на радиобутон и поле за отметка.</li> <li>– Записва синтактично и логически правилно вложени условни оператори.</li> <li>– Създава модел и алгоритъм за решаване на задача чрез използване на вложени условни конструкции.</li> <li>– Реализира модел за решаване на задача чрез използване на вложени условни оператори.</li> </ul>		Усъвършенстване на умения за логически изрази, условни конструкции, разклонени алгоритми		

			– Заменя вложен условен оператор с единичен и обратно.				
19	21. Условни конструкции и разклонени алгоритми. Обобщение	О – 1	<ul style="list-style-type: none"> <li>– Дава примери, в които се използва булев тип данни.</li> <li>– Записва на език за програмиране булев израз, съдържащ операция за сравнение.</li> <li>– Изписва синтактично правилно на езика за програмиране основните логически операции – логическо отрицание, дизюнкция, конюнкция.</li> <li>– Пресмята без използване на компютър стойността на булев израз.</li> <li>– Съставя със средствата на език за програмиране сложен булев израз, отговарящ на дадена логическа ситуация.</li> <li>Описва синтаксиса и семантиката на кратка и пълна форма на условен оператор.</li> <li>Описва разклонен алгоритъм с помощта на условен оператор.</li> <li>– Проиграва изпълнението на фрагмент на програма, съдържаща вложени условни оператори.</li> <li>– Записва синтактично и логически правилно вложени условни оператори.</li> <li>– Създава модел и алгоритъм за решаване на задача чрез използване на вложени условни конструкции.</li> <li>– Реализира модел за решаване на задача чрез използване на вложени условни оператори.</li> <li>– Заменя вложен условен оператор с единичен и обратно.</li> </ul>		Усъвършенстване на умения за условни конструкции и разклонени алгоритми.		
19 20	22. Условни конструкции и разклонени алгоритми. Практика	П – 2	<ul style="list-style-type: none"> <li>– Обосновава необходимостта от разклоняване на алгоритмичния процес.</li> <li>– Записва на език за програмиране булев израз, съдържащ операция за сравнение.</li> <li>Използва условен оператор за проверка на коректността на входните данни за програма.</li> <li>Използва условен оператор за обработка на свойства на радиобутон и поле за отметка.</li> </ul>		Усъвършенстване на умения за условни конструкции и разклонени алгоритми.		

			<ul style="list-style-type: none"> <li>– Създава модел и алгоритъм за решаване на задача чрез използване на вложени условни конструкции.</li> <li>– Реализира модел за решаване на задача чрез използване на вложени условни оператори.</li> </ul>				
20	23. Условни конструкции и разклонени алгоритми. Контрол и оценка	КО – 1	Контрол и оценка на знанията и уменията на учениците.		Контрол и оценка на знанията и уменията за условни конструкции и разклонени алгоритми.		
21	24. Програмни конструкции за реализация на циклични алгоритми	НЗУ – 2	<ul style="list-style-type: none"> <li>– Посочва елементите на циклична конструкция – инициализация, тяло на цикъла и условие на цикъла.</li> <li>– Оценява необходимостта от използването на алгоритми с циклични конструкции с предусловие и постусловие.</li> <li>– Записва синтактично и логически правилно оператори за цикъл с предусловие, постусловие и управлявани от брояч.</li> <li>– Прилага алгоритми с циклични конструкции за проверка на входни данни.</li> <li>– Оценява необходимостта от използване на алгоритми с циклични конструкции с условие или управлявани от брояч.</li> <li>– Преобразува програмен код, съдържащ циклична конструкция, управлявана от брояч, в циклична конструкция или управлявана от условие.</li> <li>– Открива синтактични и логически грешки в програмния код на алгоритми с циклична конструкция.</li> <li>– Прилага циклични алгоритми за управление на графичен потребителски интерфейс.</li> <li>– Прилага програмни конструкции за реализация на алгоритми за намиране на сума, минимален/максимален елемент, средно аритметично и др. в редици от числа, въвеждани от потребителския интерфейс/клавиатурата.</li> </ul>	<ul style="list-style-type: none"> <li>– структура на циклични алгоритмични конструкции</li> <li>– оператор за цикъл с предусловие</li> <li>– оператор за цикъл с постусловие</li> <li>– оператор за цикъл, управляван от брояч</li> <li>– списъчно поле</li> </ul>	<ul style="list-style-type: none"> <li>Въвеждане на понятията циклична алгоритмична конструкция и видове.</li> <li>Формиране на умения за оценяване на необходимостта от използването на алгоритми с циклични конструкции с предусловие и постусловие.</li> <li>Формиране на умения за записване на синтактично и логически правилно оператор за цикъл с предусловие, постусловие и с брояч.</li> <li>Формиране на умения за прилагане на алгоритми с циклични конструкции за проверка на входни данни.</li> <li>Формиране на умения за преобразуване на програмен код, съдържащ циклична конструкция, управлявана от брояч, в циклична конструкция или управлявана от условие.</li> <li>Формиране на умения за откриване на синтактични и логически грешки в програмен код на алгоритми циклична конструкция.</li> <li>Формиране на умения за прилагане на циклични алгоритми за управление на графичен потребителски интерфейс.</li> </ul>		

22	25. Програмни конструкции за реализация на циклични алгоритми. Практика	П – 2	<ul style="list-style-type: none"> <li>– Прилага циклични алгоритми за управление на графичен потребителски интерфейс.</li> <li>– Използва циклични алгоритми за въвеждане и извеждане на данни от файл.</li> </ul>		<p>Усъвършенстване на умения за използване на програмни конструкции за реализация на алгоритми за намиране на сума минимален или максимален елемент, средно аритметично и др. в редици от числа.</p>		
23	26. Програмни конструкции за реализация на циклични алгоритми в компютърна графика. Практика	П – 2	<ul style="list-style-type: none"> <li>– Прилага циклични алгоритми за изчертаване на графични примитиви.</li> </ul>	<ul style="list-style-type: none"> <li>– графичен примитив</li> <li>– кутия за изображения</li> <li>– текстов файл</li> </ul>	<p>Въвеждане на понятията графичен, примитив, кутия за изображения в текстов файл.</p> <p>Формиране на умения за прилагане на циклични алгоритми за изчертаване на графични примитиви.</p> <p>Формиране на умения за използване на циклични алгоритми за въвеждане и извеждане на данни от файл.</p>		
24	27. Циклични конструкции и циклични алгоритми. Обобщение	О – 1	<ul style="list-style-type: none"> <li>– Записва синтактично и логически правилно оператори за цикъл с предусловие, постусловие и управлявани от брояч.</li> <li>– Прилага алгоритми с циклични конструкции за проверка на входни данни.</li> <li>– Прилага програмни конструкции за реализация на алгоритми за намиране на сума, минимален/максимален елемент, средно аритметично и др. в редици от числа, въвеждани от потребителския интерфейс/клавиатурата.</li> <li>– Прилага циклични алгоритми за управление на графичен потребителски интерфейс.</li> </ul>		<p>Усъвършенстване на умения за циклични конструкции и алгоритми.</p>		
24	28. Циклични конструкции и циклични алгоритми. Контрол и оценка	КО – 1	<ul style="list-style-type: none"> <li>Контрол и оценка на знанията и уменията на учениците.</li> </ul>		<p>Контрол и оценка на знанията уменията за за циклични конструкции и алгоритми.</p>		
25	29. Тестване и верификация на програма	НЗУ – 2	<ul style="list-style-type: none"> <li>– Обяснява и разграничава понятията тестване и верификация.</li> <li>– Дефинира тестови данни</li> </ul>	<ul style="list-style-type: none"> <li>– тестване</li> <li>– верификация</li> <li>– тестови данни</li> </ul>	<p>Въвеждане на понятията тестване, верификация, тестови данни.</p>		

			<ul style="list-style-type: none"> <li>– Дефинира очаквани резултати от тестването при определени входни данни.</li> <li>– Използва инструмент за откриване и отстраняване на грешки (debugger).</li> <li>– Прилага процедури за тестване и верификация на вече създадени програми.</li> </ul>		<p>Формиране на умения за дефиниране на тестови данни и очаквани резултати от тестването при определени входни данни.</p> <p>Формиране на умения за използване на инструмент за откриване и отстраняване на грешки.</p> <p>Формиране на умения за прилагане на процедури за тестване и верификация на вече създадени програми.</p>		
26	30. Съставни типове данни. Едномерен масив	НЗУ – 2	<ul style="list-style-type: none"> <li>– Разбира необходимостта от използване на масиви.</li> <li>– Идентифицира елементите на масив.</li> <li>– Разпознава индекс и стойност на елемент на масив.</li> <li>– Дефинира масив със средствата на език за програмиране.</li> <li>– Създава и инициализира масив със средствата на език за програмиране.</li> <li>– Осъществява достъп до елемент на масив.</li> <li>– Обхожда, въвежда и извежда стойностите на елементите на масив.</li> <li>– Използва списъчно поле за извеждане на стойностите на елементите на масив.</li> </ul>	<ul style="list-style-type: none"> <li>– логическо описание на масив</li> <li>– базов тип на масив</li> <li>– индекс и стойност на елемент от масив</li> </ul>	<p>Въвеждане на понятието масив, логическо описание на масив, базов тип на масив, индекс и стойност на елемент от масив.</p> <p>Затвърждаване на знанията за елементи на масив индекс и стойност на елемент на масив.</p> <p>Формиране на умения за дефиниране на масив със средствата на език за програмиране.</p> <p>Формиране на умения за създаване и инициализиране на масив със средствата на език за програмиране.</p> <p>Формиране на умения за осъществяване на достъп до елемент на масив.</p> <p>Формиране на умения за обхождане, въвеждане и извеждане на стойностите на елементите на масив.</p> <p>Формиране на умения за използване на списъчно поле за</p>		

			<p><b>Основни алгоритми за работа с едномерен масив</b></p> <ul style="list-style-type: none"> <li>– Пресмята сбор и произведение на стойностите на елементите на едномерен масив.</li> <li>– Търси елемент от масива с максимална и минимална стойност.</li> <li>– Търси елементи от масива, отговарящи на дадено условие.</li> </ul>	<p>– последователно търсене</p>	<p>извеждане на стойностите на елементите на масив.</p> <p>Формиране на умения за пресмятане на сбор и произведение на стойностите на елементите на едномерен масив.</p> <p>Формиране на умения за търсене на елемент от масив с максимална и минимална стойност.</p> <p>Формиране на умения за търсене на елементи от масив, отговарящи на</p>		
--	--	--	--	---------------------------------	---	--	--



					дадено условие.		
27	31. Съставни типове данни. Едномерен масив. Практика	П – 2	<ul style="list-style-type: none"> <li>– Дефинира масив със средствата на език за програмиране.</li> <li>– Създава и инициализира масив със средствата на език за програмиране.</li> <li>– Осъществява достъп до елемент на масив.</li> <li>– Обхожда, въвежда и извежда стойностите на елементите на масив.</li> <li>– Използва списъчно поле за извеждане на стойностите на елементите на масив.</li> </ul>		Усъвършенстване на умения за работа с едномерни масиви.		
28	32. Обработка на редици от елементи. Обобщение	О – 1	<ul style="list-style-type: none"> <li>– Разбира необходимостта от използване на масиви.</li> <li>– Дефинира масив със средствата на език за програмиране.</li> <li>– Създава и инициализира масив със средствата на език за програмиране.</li> <li>– Осъществява достъп до елемент на масив.</li> <li>– Обхожда, въвежда и извежда стойностите на елементите на масив.</li> <li>– Използва списъчно поле за извеждане на стойностите на елементите на масив.</li> <li>– Пресмята сбор и произведение на стойностите на елементите на едномерен масив.</li> </ul>		Усъвършенстване на умения за работа с редици от елементи.		
			<ul style="list-style-type: none"> <li>– Търси елемент от масива с максимална и минимална стойност.</li> <li>– Търси елементи от масива, отговарящи на дадено условие.</li> </ul>				
28 29	33. Обработка на редици от елементи. Практика	П – 2			Усъвършенстване на умения по обработка на редица от елементи.		
29 30	34. Общи задачи по програмиране. Практика	П – 2			Усъвършенстване на умения по типове данни, условни конструкции, циклични алгоритми и едномерни масиви.		
30	35. Обработка на редици от елементи. Контрол и оценка	КО – 1	Контрол и оценка на знанията и уменията на учениците		Контрол и оценка на знанията и уменията за редици от елементи.		

**СОФТУЕРЕН ПРОЕКТ**

31	36. Създаване на софтуерен проект	НЗУ – 2	<ul style="list-style-type: none"> <li>– Описва етапите при реализиране на софтуерен проект.</li> <li>– Извършва проучване и анализ на решения за даден групов проект.</li> <li>– Създава модел за решаване на проблема, поставен в заданието на проекта.</li> <li>– Проектира графичен потребителски интерфейс.</li> <li>– Създава програмен код за реализация на модела.</li> <li>– Създава тестови примери с входни данни и очаквани резултати.</li> <li>– Изготвя документация за софтуерния проект.</li> <li>– Презентира и защитава готовия софтуерен проект.</li> </ul>		<p>Въвеждане на етапите при реализиране на софтуерен проект.</p> <p>Запознаване с начини за проучване и анализ на решения за даден групов проект.</p> <p>Формиране на умения за създаване на модел за решаване на проблема поставяне заданието на проекта.</p> <p>Формиране на умения за проектиране на графичен потребителски интерфейс.</p> <p>Актуализиране на знания и умения за създаване на програмен код за реализация на модела.</p> <p>Актуализиране на знания и умения за създаване на тестови примери с входни данни и очаквани резултати.</p> <p>Формиране на умения за изготвяне на документация за собствен проект</p>		
					<p>Актуализиране на знания за презентиране и защита на готов софтуерен проект</p>		

32	37. Създаване на софтуерен проект. Практика	П – 2	<ul style="list-style-type: none"> <li>– Описва етапите при реализиране на софтуерен проект.</li> <li>– Извършва проучване и анализ на решения за даден групов проект.</li> <li>– Създава модел за решаване на проблема, поставен в заданието на проекта.</li> <li>– Проектира графичен потребителски интерфейс.</li> <li>– Създава програмен код за реализация на модела.</li> <li>– Създава тестови примери с входни данни и очаквани резултати.</li> </ul>		<p>Актуализиране на знания за етапите при реализиране на софтуерен проект.</p> <p>Усъвършенстване на умения за създаване на модел за решаване на проблема.</p> <p>Усъвършенстване на умения за създаване на програмен код за реализация на модела.</p> <p>Усъвършенстване на умения за създаване на тестови примери с входни данни и очаквани резултати.</p>		
33	38. Създаване на софтуерен проект. Практика	П – 2	<ul style="list-style-type: none"> <li>– Извършва проучване и анализ на решения за даден групов проект.</li> <li>– Създава модел за решаване на проблема, поставен в заданието на проекта.</li> <li>– Проектира графичен потребителски интерфейс.</li> <li>– Създава програмен код за реализация на модела.</li> <li>– Създава тестови примери с входни данни и очаквани резултати.</li> </ul>		<p>Запознаване с начини за проучване и анализ на решения за даден групов проект.</p> <p>Формиране на умения за създаване на модел за решаване на проблема за поставяне на заданието на проекта.</p> <p>Формиране на умения за проектиране на графичен потребителски интерфейс.</p> <p>Актуализиране на знания и умения за създаване на програмен код за реализация на модела.</p> <p>Актуализиране на знания и умения за създаване на тестови примери с входни данни и очаквани резултати.</p>		
34	39. Създаване на софтуерен проект. Практика	П – 2	<ul style="list-style-type: none"> <li>– Описва етапите при реализиране на софтуерен проект.</li> <li>– Извършва проучване и анализ на решения за даден групов проект.</li> </ul>		<p>Усъвършенстване на умения за познаване и прилагане на етапите при реализиране на софтуерен проект.</p>		

			<ul style="list-style-type: none"> <li>– Създава модел за решаване на проблема, поставен в заданието на проекта.</li> <li>– Създава програмен код за реализация на модела.</li> <li>– Създава тестови примери с входни данни и очаквани резултати.</li> <li>– Изготвя документация за софтуерния проект.</li> </ul>		<p>Актуализиране на умения за провеждане на проучвания на необходимост от решения за даден проблем.</p> <p>Усъвършенстване на умения за създаване на модел за решаване на проблема.</p> <p>Усъвършенстване на умения за създаване на програмен код за реализация на модела.</p> <p>Актуализиране на знания и умения за създаване на тестови примери с входни данни и очаквани резултати.</p> <p>Формиране на умения за изготвяне на документация за собствен проект.</p>		
35	40. Създаване на софтуерен проект. Практика	П – 2	<ul style="list-style-type: none"> <li>– Описва етапите при реализиране на софтуерен проект.</li> <li>– Извършва проучване и анализ на решения за даден групов проект.</li> <li>– Създава модел за решаване на проблема, поставен в заданието на проекта.</li> <li>– Създава програмен код за реализация на модела.</li> <li>– Създава тестови примери с входни данни и очаквани резултати.</li> <li>– Изготвя документация за софтуерния проект.</li> <li>– Презентира и защитава готовия софтуерен проект.</li> </ul>		<p>Усъвършенстване на умения за познаване и прилагане на етапите при реализиране на софтуерен проект.</p> <p>Актуализиране на умения за провеждане на проучвания на необходимост от решения за даден проблем.</p> <p>Усъвършенстване на умения за създаване на модел за решаване на проблема.</p> <p>Усъвършенстване на умения за създаване на програмен код за реализация на модела.</p> <p>Актуализиране на знания и умения за създаване на тестови примери с входни данни и очаквани резултати.</p>		
					<p>Формиране на умения за изготвяне на документация за собствен проект.</p>		

36	41. Създаване на софтуерен проект. Контрол и оценка	КО – 1	Контрол и оценка на знанията и уменията на учениците по темата „Създаване на софтуерен проект“.		Контрол и оценка на знанията и уменията за софтуерен проект.		
36	42. Проверка на знанията и уменията по информатика. Изходно равнище	КО – 1	Контрол и оценка на знанията и уменията на учениците по информатика.		Контрол и оценка на знанията и уменията по информатика.		