

## **КОНТРОЛЬНО - ОЦЕНОЧНЫЕ СРЕДСТВА**

### **промежуточной аттестации**

Наименование учебного предмета: информатика

Класс (ы): 12

Форма обучения: очно-заочная

Уровень общего образования: среднее общее образование

Учитель: Щипанова В.А.

### **Промежуточная аттестация по курсу информатика. 10 класс.**

**Цель:** контроль результатов обучения учебной деятельности:

#### **ЛИЧНОСТНЫЕ РЕЗУЛЬТАТЫ**

##### **1) гражданского воспитания:**

осознание своих конституционных прав и обязанностей, уважение закона и правопорядка, соблюдение основополагающих норм информационного права и информационной безопасности;

готовность противостоять идеологии экстремизма, национализма, ксенофобии, дискриминации по социальным, религиозным, расовым, национальным признакам в виртуальном пространстве;

##### **2) патриотического воспитания:**

ценностное отношение к историческому наследию, достижениям России в науке, искусстве, технологиях, понимание значения информатики как науки в жизни современного общества;

##### **3) духовно-нравственного воспитания:**

сформированность нравственного сознания, этического поведения; способность оценивать ситуацию и принимать осознанные решения, ориентируясь на морально-нравственные нормы и ценности, в том числе в сети Интернет;

##### **4) эстетического воспитания:**

эстетическое отношение к миру, включая эстетику научного и технического творчества;

способность воспринимать различные виды искусства, в том числе основанные на использовании информационных технологий;

##### **5) трудового воспитания:**

готовность к активной деятельности технологической и социальной направленности, способность инициировать, планировать и самостоятельно выполнять такую деятельность;

интерес к сферам профессиональной деятельности, связанным с информатикой, программированием и информационными технологиями, основанными на достижениях информатики и научно-технического прогресса, умение совершать осознанный выбор будущей профессии и реализовывать собственные жизненные планы;

готовность и способность к образованию и самообразованию на протяжении всей жизни;

##### **6) экологического воспитания:**

осознание глобального характера экологических проблем и путей их решения, в том числе с учётом возможностей информационно-коммуникационных технологий;

##### **7) ценности научного познания:**

сформированность мировоззрения, соответствующего современному уровню развития информатики, достижениям научно-технического прогресса и общественной практики, за счёт понимания роли информационных ресурсов, информационных процессов и информационных технологий в условиях цифровой трансформации многих сфер жизни современного общества;

осознание ценности научной деятельности, готовность осуществлять проектную и

исследовательскую деятельность индивидуально и в группе.

В процессе достижения личностных результатов освоения программы по информатике у обучающихся совершенствуется эмоциональный интеллект, предполагающий сформированность:

саморегулирования, включающего самоконтроль, умение принимать ответственность за своё поведение, способность адаптироваться к эмоциональным изменениям и проявлять гибкость, быть открытым новому;

внутренней мотивации, включающей стремление к достижению цели и успеху, оптимизм, инициативность, умение действовать, исходя из своих возможностей;

эмпатии, включающей способность понимать эмоциональное состояние других, учитывать его при осуществлении коммуникации, способность к сочувствию и сопереживанию;

социальных навыков, включающих способность выстраивать отношения с другими людьми, заботиться, проявлять интерес и разрешать конфликты.

## **МЕТАПРЕДМЕТНЫЕ РЕЗУЛЬТАТЫ**

### **Познавательные универсальные учебные действия**

#### **Базовые логические действия:**

самостоятельно формулировать и актуализировать проблему, рассматривать её всесторонне;

устанавливать существенный признак или основания для сравнения, классификации и обобщения;

определять цели деятельности, задавать параметры и критерии их достижения; выявлять закономерности и противоречия в рассматриваемых явлениях; разрабатывать план решения проблемы с учётом анализа имеющихся материальных и нематериальных ресурсов;

вносить коррективы в деятельность, оценивать соответствие результатов целям, оценивать риски последствий деятельности;

координировать и выполнять работу в условиях реального, виртуального и комбинированного взаимодействия;

развивать креативное мышление при решении жизненных проблем.

#### **Базовые исследовательские действия:**

владеть навыками учебно-исследовательской и проектной деятельности, навыками разрешения проблем, способностью и готовностью к самостоятельному поиску методов решения практических задач, применению различных методов познания;

овладеть видами деятельности по получению нового знания, его интерпретации, преобразованию и применению в различных учебных ситуациях, в том числе при создании учебных и социальных проектов;

формирование научного типа мышления, владение научной терминологией, ключевыми понятиями и методами;

ставить и формулировать собственные задачи в образовательной деятельности и жизненных ситуациях;

выявлять причинно-следственные связи и актуализировать задачу, выдвигать гипотезу её решения, находить аргументы для доказательства своих утверждений, задавать параметры и критерии решения;

анализировать полученные в ходе решения задачи результаты, критически оценивать их достоверность, прогнозировать изменение в новых условиях;

давать оценку новым ситуациям, оценивать приобретённый опыт; осуществлять целенаправленный поиск переноса средств и способов действия

в профессиональную среду;

переносить знания в познавательную и практическую области жизнедеятельности; интегрировать знания из разных предметных областей;

выдвигать новые идеи, предлагать оригинальные подходы и решения, ставить проблемы и задачи, допускающие альтернативные решения.

### **Работа с информацией:**

владеть навыками получения информации из источников разных типов, самостоятельно осуществлять поиск, анализ, систематизацию и интерпретацию информации различных видов и форм представления;

создавать тексты в различных форматах с учётом назначения информации и целевой аудитории, выбирая оптимальную форму представления и визуализации;

оценивать достоверность, легитимность информации, её соответствие правовым и морально-этическим нормам;

использовать средства информационных и коммуникационных технологий в решении когнитивных, коммуникативных и организационных задач с соблюдением требований эргономики, техники безопасности, гигиены, ресурсосбережения, правовых и этических норм, норм информационной безопасности;

владеть навыками распознавания и защиты информации, информационной безопасности личности.

### **Коммуникативные универсальные учебные действия**

#### **Общение:**

осуществлять коммуникации во всех сферах жизни;

распознавать невербальные средства общения, понимать значение социальных знаков, распознавать предпосылки конфликтных ситуаций и уметь смягчать конфликты;

владеть различными способами общения и взаимодействия, аргументированно вести диалог;

развёрнуто и логично излагать свою точку зрения.

#### **Совместная деятельность:**

понимать и использовать преимущества командной и индивидуальной работы;

выбирать тематику и методы совместных действий с учётом общих интересов и возможностей каждого члена коллектива;

принимать цели совместной деятельности, организовывать и координировать действия по её достижению: составлять

план действий, распределять роли с учётом мнений участников, обсуждать результаты совместной работы;

оценивать качество своего вклада и каждого участника команды в общий результат по разработанным критериям;

предлагать новые проекты, оценивать идеи с позиции новизны, оригинальности, практической значимости;

осуществлять позитивное стратегическое поведение в различных ситуациях, проявлять творчество и воображение, быть инициативным.

### **Регулятивные универсальные учебные действия**

#### **Самоорганизация:**

самостоятельно осуществлять познавательную деятельность, выявлять проблемы, ставить и формулировать собственные задачи в образовательной деятельности и жизненных ситуациях;

самостоятельно составлять план решения проблемы с учётом имеющихся ресурсов, собственных возможностей и предпочтений;

давать оценку новым ситуациям;

расширять рамки учебного предмета на основе личных предпочтений;

делать осознанный выбор, аргументировать его, брать ответственность за решение;

оценивать приобретённый опыт;

способствовать формированию и проявлению широкой эрудиции в разных областях знаний, постоянно повышать свой образовательный и культурный уровень.

**Самоконтроль:**

давать оценку новым ситуациям, вносить коррективы в деятельность, оценивать соответствие результатов целям;

владеть навыками познавательной рефлексии как осознания совершаемых действий и мыслительных процессов, их результатов и оснований; использовать приёмы рефлексии для оценки ситуации, выбора верного решения;

оценивать риски и своевременно принимать решения по их снижению; принимать мотивы и аргументы других при анализе результатов деятельности.

**Принятия себя и других:**

принимать себя, понимая свои недостатки и достоинства;

принимать мотивы и аргументы других при анализе результатов деятельности;

признавать своё право и право других на ошибку;

развивать способность понимать мир с позиции другого человека.

**ПРЕДМЕТНЫЕ РЕЗУЛЬТАТЫ**

- умение реализовывать на выбранном для изучения языке программирования высокого уровня (Паскаль, Python, Java, C++, C#) типовые алгоритмы обработки чисел, числовых последовательностей и массивов: представление числа в виде набора простых сомножителей, нахождение максимальной (минимальной) цифры натурального числа, записанного в системе счисления с основанием, не превышающим 10, вычисление обобщённых характеристик элементов массива или числовой последовательности (суммы, произведения, среднеарифметического, минимального и максимального элементов, удовлетворяющему заданному условию), сортировку элементов массива;

- умение использовать табличные (реляционные) базы данных, в частности, составлять запросы к базам данных (в том числе запросы с вычисляемыми полями), выполнять сортировку и поиск записей в базе данных, наполнять разработанную базу данных, умение использовать электронные таблицы для анализа, представления обработки данных (включая вычисление суммы, среднего арифметического, наибольшего и наименьшего значений, решение уравнений);

- умение использовать компьютерно – математические модели для анализа объектов и процессов: формулировать цель моделирования, выполнять анализ результатов, полученных в ходе моделирования, оценивать соответствие модели моделируемому объекту или процессу, представлять результаты моделирования в наглядном виде;

- умение организовывать личное информационное пространство с использованием различных цифровых технологий, понимание возможностей цифровых сервисов государственных услуг, цифровых образовательных сервисов, понимание возможностей и ограничений технологий искусственного интеллекта в различных областях, наличие представлений об использовании информационных технологий в различных профессиональных сферах.

*Внимательно прочитайте текст.*

**Алгоритмы и программирование.**

**Обработка символьных данных.** Символьный тип (char) – это множество символов кодовой таблицы компьютера ASCII. Значением переменной символьного типа может быть любой отображаемый символ:

- буква русского или латинского алфавитов;

- цифра;

- знак препинания;

Переменная символьного типа должна быть объявлена в разделе объявления переменных. var : char; В тексте программы значения переменных и константы символьного типа должны быть заключены в апострофы: 'A', 'C', '+'

Как и любая переменная программы, переменная типа `char` может получить значение в результате выполнения оператора присваивания. `c1 := '*';` Над данными символьного типа определены следующие операции отношения: `=`, `<>`, `,`, `<=`, `=>`, вырабатывающие результат логического типа. Переменную типа `char` можно сравнить с другой переменной типа `char` или с символьной константой. Сравнение основано на том, что каждому символу поставлено в соответствие число из кодовой таблицы ASCII. '0'<'1'<...<'9'<...<'A'<'B'<...<'Z'<'a'<'b'<...<'z' Для данных символьного типа определены следующие **стандартные функции**:

- `Chr(x)` - возвращает символ, соответствующий ASCII-коду числа `X`. Выражение Результат `Chr(65)` 'A'
- `Ord(ch)`- возвращает число, соответствующее символу `ch` в ASCII-таблице. Выражение Результат `Ord('A')` 65
- `Pred(ch)`- возвращает символ, который предшествует в ASCII-таблице символу `ch`. Выражение Результат `Pred('B')` 'A'
- `Succ(ch)`- возвращает символ, который следует в ASCII-таблице за символом `ch`. Выражение Результат `Succ('A')` 'B'

Описание строкового типа. Строка - это последовательность символов кодовой таблицы ASCII. При использовании в выражениях строка заключается в апострофы. Количество символов в строке может изменяться динамически от 0 до 255. Для определения данных строкового типа используется идентификатор `string`, за которым следует заключенное в квадратные скобки значение максимально допустимой длины строки данного типа. Если это значение не указывается, то по умолчанию длина строки принимается равной 255 байт.

**Строковые выражения.** Выражения, в которых операторами служат строковые данные, называются строковыми. Они состоят из строковых переменных, констант, указателей функций и знаков операций. Для присвоения строковой переменной результата строкового выражения используют оператор присвоения (`:=`). `St1 := 'Иванов И.И.'`

Индекс определяется выражением целочисленного типа, которое записывается в квадратных скобках сразу за идентификатора строковой переменной или константы. Например: `St1 := 'Иванов';` Выражение `St1[5]` обеспечивает доступ к пятому символу 'о'.

Строковые процедуры и функции. Функции:

- `Length(st)` – определяет текущую длину строки `st`. Результат имеет целочисленный тип.
- `Copy(St, Poz, N)` – выделяет из строки `St` `N` символов, начиная с символа с номером `Poz`. Если `Poz>Length(st)`, то результатом будет ноль. Если `Poz>255`, то возникает ошибка. `Poz` и `N` целочисленные числа.
- `Concat(St1, St2, ... , StN)` – выполняет сцепление строк `St1`, `St2`, ..., `StN` в том порядке, в котором они указаны в списке параметров.
- `Pos (St1, St2)` – определяет номер символа, с которого начинается вхождение подстроки `St1` в строку `St2`. Результат целочисленного типа. Если подстрока не найдена, то результат ноль.
- `Ucase (ch)` – преобразует строчную букву в прописную. Обрабатываются только буквы латинского алфавита. Если значением `ch` является любой другой символ (в том числе строчная буква русского алфавита), функция возвращает его без преобразования.

Процедуры:

- `Delete (st, Poz, N)` - удаляет `N` символов начиная с позиции `Poz` из строки `st`. Если значение `Poz>255`, возникает программное прерывание.
- `Insert (St1, St2, Poz)` - вставляет подстроку `St1` в строку `St2`, начиная с позиции `Poz`.
- `Str(X, St)` – преобразует числовое значение в строку цифр. `X` – преобразуемое число, `St` – возвращаемая строка. После `X` можно записать формат, аналогично формату вывода. Если в формате указано недостаточное для вывода количество разрядов, поле вывода автоматически расширяется до нужной длины.
- `Val(St, X, code)` - преобразует значение `St` в величину целочисленного или вещественного типа и помещает

A[1]	A[2]	...	A[N]
------	------	-----	------

  

A[1, 1]	A[1, 2]	...	A[1, N]
A[2, 1]	A[2, 2]	...	A[2, N]
...	...	...	...
A[M, 1]	A[M, 2]	...	A[M, N]

результат в переменную X. Значение St не должно содержать незначащих пробелов в конце и в начале. Code – целочисленная переменная. Если операция преобразования не привела к ошибке, то значение Code равно нулю. В случае ошибки Code будет содержать номер позиции первого ошибочного символа, а значение X будет неопределенно. Табличные величины (массивы).

В практической деятельности человек часто использует всевозможные таблицы. Это, например, список учащихся в классном журнале, табель успеваемости, таблица результатов спортивных соревнований и т. д. Чаще всего встречаются линейные и прямоугольные таблицы.

**Линейная таблица.** Линейная таблица (одномерный массив) представляет собой набор однотипных данных, записанных в одну строку или один столбец. Элементы строки (столбца) всегда нумеруются.

Прямоугольная таблица (двумерный массив) — это упорядоченный некоторым образом набор строк (столбцов), содержащих одинаковое количество элементов. Строки прямоугольных таблиц имеют свою нумерацию, столбцы — свою.

*Пользуясь текстом, ответьте на вопросы:*

1. Какие стандартные функции определены для данных символьного типа?
2. Что такое строка?
3. Что такое линейная таблица?
4. Что такое прямоугольная таблица?
5. Введите строку и подсчитайте количество слов в ней. Слова разделяются пробелами.

#### **Эталоны ответов.**

1	<b>chr(x).</b> Преобразует выражение x типа byte в символ и возвращает значение символа. <u>1</u> <b>ord(ch).</b> Преобразует символ ch в его код типа byte и возвращает значение кода. <u>1</u> <b>pred(ch).</b> Возвращает предыдущий символ. <u>1</u> <b>succ(ch).</b> Возвращает следующий символ. <u>1</u> <b>upcase(ch).</b> Преобразует строчную букву в заглавную.
2	Строка или строковое выражение - это последовательность символов либо в виде литеральной константы, либо в виде какой-либо переменной.
3	Линейная таблица (одномерный массив) представляет собой набор однотипных данных, записанных в одну строку или один столбец.
4	Прямоугольная таблица (двумерный массив) — это упорядоченный некоторым образом набор строк (столбцов), содержащих одинаковое количество элементов.
5	Обозначим: St – исходная строка; L – длина исходной строки; K – количество слов в строке. Для подсчета количества слов необходимо организовать цикл. Так как после слова может стоять не один пробел, а несколько, то для определения слова используется условие (St[I]<>' ') AND (St[I+1]=' '). Для того чтобы при подсчете количества слов учесть последнее слово, в конец введенной строки добавляется пробел: St:=St + ' '. <i>Программа</i> Program Prim211; Var St : String; I, K, L : Integer;

	Begin Writeln('Введите строку'); Readln(St); L := Length(St); St:=St + ' '; K := 0; For I:=1 To L Do If (St[I]<>' ') AND (St[I+1]=' ') Then K:=K+1; Writeln('Количество слов = ', K); End.
--	---

### **Критерии оценивания**

- Оценка «5» - выполнены все задания;
- Оценка «4» - выполнены 4 задания;
- Оценка «3» - выполнены 3 задания;
- Оценка «2» - выполнены меньше трех заданий;
- Оценка «1» - выполнены меньше двух заданий.