**Задача:** по коду символа, введенного с клавиатуры, определить, является

этот символ цифрой, буквой латинского либо русского алфавита или другим

символом.

Вывести в консоль символ, информацию о принадлежности символа к одной

из категорий, его код в соответствующей кодировке ASCII или

Windows- 1251.

Программа принимает исходный символ. Проверяет код символа на соответствие поставленным условиям. Выводит соответствующую подходящему условию фразу в окно вывода.

**Словесно-формульный способ**

1. Ввести символ.

2. Если код символа попадает в диапазон от 3016 до 3916 включительно, то п.3, в противном случае п.5.

3. Вывести «Это цифра», символ цифры, ASCII, код символа в таблице ASCII.

4. Перейти к п.12 (конец).

5. Иначе: если код символа попадает в диапазон от 4116 до 7A16

включительно, то п.6, в противном случае п.8.

6. Вывести «Это латинская буква», символ буквы, ASCII, код символа в таблице ASCII.

7. Перейти к п.12 (конец).

8. Иначе: если код символа попадает в диапазон от 0xC0 до 0xFF

включительно, то п.9 в противном случае п.11.

9. Вывести «Это русская буква», символ буквы, Windows- 1251, код символа в таблице Windows- 1251.

10. Перейти к п.12 (конец).

11. Вывести «Это не цифра и не буква», символ, код символа в таблице Windows- 1251.

12. КОНЕЦ.

**Псевдокод**

НАЧАЛО

ВВОД символа

ЕСЛИ это цифра ТО

ВЫВОД «это цифра»

ЕСЛИ это латинская буква ТО ВЫВОД «это латинская буква»

ЕСЛИ это русская буква ТО ВЫВОД «это русская буква»

ИНАЧЕ

ВЫВОД «это не цифра и не буква»

КОНЕЦ

**Блок-схема**

Начало

Не цифра, не буква

Это русская буква

Это латинская буква

Это цифра

Конец

Ввод

Это цифра?

Это л.бук?

Это р.бук?

да

нет

да

нет

да

нет

1. Для создания нового проекта запустите Visual Studio.
2. Нажмите создать проект
3. Выберите консольное приложение c++
4. Введите имя и расположение проекта
5. Нажмите создать

Начало

Конец

Создать

Ввести имя и расположения

Выбрать консольные приложения

Создать проект

Разница в 3 цифре номера

PUCHKO(0050 0055 0043 0048 004B 004F)

Puchko(0070 0075 0063 0068 006B 006F)

Чтобы перевести прописную букву в строчную нужно прибавить 2 к 3 цифре номера символа.

Начало

Конец

Вывести результат

Это не прописные

Прописные?

Кодировка в

Windows-1251

Прибавить к каждому номеру 20

раскодировать

Ввод

**Алгоритм** - это определенная последовательность (порядок) действий, строгое выполнение которых приведет к выполнению цели.

1. **Массовость** (универсальность). Благодаря этому свойству, алгоритм можно успешно применять к различным наборам исходных данных. Пусть существует запись некой абстрактной последовательности, выраженная формулой. Подставляя в эту формулу значения (каждый раз новые), пользователь будет получать верные решения в соответствии с определенным алгоритмом действий.

2. **Дискретность** (разрывность). Это свойство характеризует структуру. Каждая алгоритмическая последовательность действий делится на этапы (шаги), а процесс решения задачи — это последовательное исполнение простых шагов. Также дискретность обозначает, что для выполнения каждого этапа потребуется конечный временной отрезок (исходные данные преобразуются во времени в результат **дискретно**).

3. **Определенность** (точность, детерминированность) — это свойство указывает алгоритму, что каждый его шаг должен быть строго определенным, то есть различные толкования должны быть исключены. Строго определяется и порядок выполнения шагов. В результате каждый шаг определяется состоянием системы однозначно, когда четко понятно, какая команда станет выполняться на следующем шаге. Как итог — при любом исполнителе для одних и тех же исходных данных при выполнении одной и той же цепочки команд будет выдаваться одинаковый результат. Да, существуют вероятностные алгоритмы — в них на последующий шаг влияют как текущее состояние системы, так и генерируемое случайное число. Но при включении способа генерации рандомных чисел в перечень «исходных данных», вероятностный алгоритм превращается в подвид обычного.

4. **Понятность**. Должны быть включены лишь те команды, которые доступны и понятны исполнителю, то есть входят в систему его команд.

5. **Формальность**. Любой исполнитель действует формально и лишь выполняет инструкции, не вникая в смысл. Он не отвлекается от поставленной задачи и не рассуждает, зачем и почему они нужны. Рассуждениями занимается разработчик алгоритма, задача же исполнителя — просто исполнить предложенные команды и получить результат. «Приказы не обсуждают, а выполняют».

6. **Завершаемость** (конечность). Если исходные данные заданы корректно, алгоритм завершит свое действие и выдаст результат за конечное число шагов.

7. **Результативность**. Согласно этому свойству, любой алгоритм должен завершаться конкретными результатам

вербальный, когда алгоритм описывается на человеческом языке

символьный, когда алгоритм описывается с помощью набора символов

графический, когда алгоритм описывается с помощью набора графических изображений

**линейный** – список команд (указаний), выполняемых последовательно друг за другом; разветвляющийся – алгоритм, содержащий хотя бы одну проверку условия, в результате которой обеспечивается переход на один из возможных вариантов решения.

циклический – алгоритм, предусматривающий многократное повторение одной  и  той  же  последовательности  действий.  Количество  повторений  обусловливается исходными данными или условием задачи.