**Nanjing University of Posts and Telecommunications**



Course: Coding Theory and Application of Chinese Character

School：计算机学院、软件学院、网络空间安全学院

Student No: F1225040103

Name: Mahim Al Muntashir Billah

Submission date: 2025-mm-dd

Lecturer: Yimu Ji / Shangdong Liu

Major: MCT

|  |  |
| --- | --- |
| Course Goals / Evaluation Criteria | Grade |
| **Course Goal 1：** Understanding the development history of computer Chinese character coding and understanding the difficulties of computer Chinese character coding; understanding the problems that pinyin faces when foreigners learn Chinese. (20 Points) | 16 |
| **Course Goal 2：** Master the concept of Chinese character components, master the concept of Chinese radicals and its meaning in Chinese character formation. (20 Points) | 16 |
| **Course Goal 3：**To understand the relationship between the stroke and the root of the radical. The general differences and special cases in the number of strokes (single strokes), know how to count the number of strokes of more than 200 commonly used Chinese radicals, and understand the subtle differences between strokes and composite strokes. (20 Points) | 16 |
| **Course Goal 4：**Design a Chinese input method and provide the flowcharts. (20 Points) | 16 |
| **Course Goal 5：**Implement a Chinese input method software. (20 Points) | 16 |
| **Totally** | 80 |
| Comments:    2025-mm-dd | |

# 1. Chinese character coding

### ****1.1 Overview of Chinese Character Coding****

Chinese character encoding refers to the systematic process of converting Chinese characters into a digital format that computers can understand. Due to the large number of Chinese characters (approximately 50,000 to 100,000), encoding systems need to efficiently map each character to a unique binary representation, while ensuring compatibility across various software and hardware platforms.

The primary goals of Chinese character encoding systems include:

* **Representation**: Accurate digital representation of characters.
* **Efficiency**: Ensuring compact encoding to save memory and reduce transmission time.
* **Compatibility**: Allowing Chinese characters to be used seamlessly in computing, including integration with international encoding standards like ASCII.

Before the digital era, Chinese characters were often represented using telegraph codes, which used numbers to correspond to characters. However, these early methods lacked standardization and had significant limitations in terms of character coverage and cross-platform compatibility. As computing needs grew, standardized encoding systems for Chinese characters became increasingly important to facilitate widespread adoption of computer technology in China and globally.

### ****1.2 Historical Development of Chinese Character Coding****

The development of Chinese character encoding has evolved significantly since the early days of computing. Over time, several standards have been introduced to handle the complexities of encoding Chinese characters, especially given the vast range of symbols in Chinese writing.

**Telegraph Codes (Pre-1980s):** Before the widespread use of computers, early attempts to digitally transmit Chinese characters used telegraph codes, where each character was assigned a numeric value. This method was cumbersome and limited by the inability to handle the large number of characters in the Chinese language.

**GB2312-80 (1980):** The **GB2312-80** standard was the first national encoding system for Simplified Chinese characters in mainland China. It introduced a two-byte encoding scheme that could represent 6,763 characters and 682 symbols, allowing compatibility with ASCII for English characters. This was a breakthrough for the digital representation of Chinese text, providing a uniform framework for Chinese character processing and communication.

**Big5 (1984):** The **Big5** encoding standard was developed in Taiwan and Hong Kong for Traditional Chinese characters. It utilized a two-byte encoding scheme similar to GB2312 but could represent up to 13,053 characters. Big5 became the de facto encoding standard for Traditional Chinese and remains in use for specific regions, especially in Taiwan and Hong Kong.

**Unicode (1991): Unicode** introduced a universal character set that aimed to cover all written languages, including Chinese. Unicode supports both Simplified and Traditional Chinese characters and offers various encoding schemes such as **UTF-8**, **UTF-16**, and **UTF-32**. UTF-8 has become the most widely used encoding for Chinese text on the internet, offering efficient storage for characters in variable-length encoding.

**GBK (1993):** The **GBK** encoding system, an extension of GB2312, was developed to support 21,003 characters, including those from ethnic minority languages. It preserved backward compatibility with GB2312, using a two-byte encoding for characters. This extension further enhanced the digital representation of Chinese characters and bridged gaps between Simplified and Traditional Chinese scripts.

**GB18030 (2000–2005):** The **GB18030** encoding standard is the most comprehensive national standard for Chinese characters, supporting over 70,000 characters. It uses a variable-length encoding scheme, similar to Unicode's UTF-8, with one to four bytes per character. This standard ensures the complete coverage of Chinese characters, including those used by minority languages, and is still enforced in mainland China.

### ****1.3 Modern Standards and Significance****

Today, **Unicode** is the predominant encoding system worldwide, ensuring that Chinese characters are correctly represented across all modern software and platforms. In China, **GB18030** is the official national encoding standard, ensuring compatibility with international systems like Unicode while maintaining support for regional and minority languages.

The evolution of Chinese character encoding has been instrumental in:

* **Digital Communication:** Enabling the seamless transmission of Chinese text over the internet, emails, and digital documents.
* **Cross-Language Integration:** Supporting the integration of Chinese characters with other languages in global computing systems.
* **Technological Development:** Facilitating the development of Chinese input methods (IMEs) and other software systems tailored to the unique needs of Chinese users.

## ****2. Chinese Character Components****

### ****2.1 Overview of Character Components****

Chinese characters are not simple visual symbols; they are made up of smaller, meaningful units called **components**. These components play a vital role in understanding how characters are formed and how they can be input efficiently into computer systems. Components include **radicals**, **phonetic elements**, and **structural parts**. Together, these components provide meaning, aid pronunciation, and contribute to the overall structure of the character.

Understanding these components is key for both **language learners** and **Chinese input method designers**. By breaking down characters into their components, learners can more easily remember and recognize them, while input method designers can create more efficient systems based on predictable patterns.

### ****2.2 Types of Components****

Chinese characters are composed of various types of components, each playing a different role in the character’s construction:

**Radicals (部首, bùshǒu):** Radicals are the key components of Chinese characters, often providing semantic clues about the meaning of the character. Radicals are also used as the primary indexing mechanism in Chinese dictionaries. For example:

* The radical **水 (氵)** signifies "water" and is found in characters related to water like **河 (hé, river)** and **海 (hǎi, sea)**.
* The radical **木** represents "wood" and is used in characters like **林 (lín, forest)** and **森 (sēn, woods)**.

**Phonetic Components:** These components offer clues to the pronunciation of the character. They are often combined with radicals to form a complete character. For example, **妈 (mā)** consists of the radical **女 (nǚ, woman)** and the phonetic component **马 (mǎ, horse)**. The **马** component helps give the character its pronunciation.

**Compound Components:** Some characters are made up of compounds that serve both phonetic and semantic functions. For example, **明 (míng, bright)** consists of the radicals **日 (rì, sun)** and **月 (yuè, moon)**, conveying the meaning of light, with **月** also serving a phonetic role.

**Structural Components:** Structural components refer to the patterns that make up the visual form of the character, regardless of its meaning. These components help define how a character is written, affecting its aesthetic and balance. For instance, characters like **口 (kǒu, mouth)** are simple, composed of a single component, whereas others like **国 (guó, country)** have more complex structures.

**Single-Component Characters:** Some characters consist of a single, indivisible component that represents a complete meaning, such as **一 (yī, one)** or **木 (mù, wood)**. These characters are simple and do not require further decomposition into smaller components.

**Recurrent Components:** Certain components appear frequently in multiple characters. For example, **心 (xīn, heart)** appears in characters like **忙 (máng, busy)** and **想 (xiǎng, think)**, which are related to emotions or thought processes.

**Pictographic Components:** The earliest forms of Chinese characters were pictographs, where the component directly depicted an object. For example, the character **山 (shān, mountain)** is a simplified representation of a mountain, and **日 (rì, sun)** represents the sun.

### ****2.3 Example****

One of the most famous examples of Chinese character components is the character **好 (hǎo, good)**, which is composed of two components: **女 (nǚ, woman)** and **子 (zǐ, child)**. This character represents the concept of good, as traditionally, the combination of a woman and child was seen as an ideal representation of goodness.

By understanding these components and their functions, it becomes easier to break down and understand even the most complex Chinese characters.

### ****2.4 Stroke Type and Stroke Amount****

**Strokes** are the building blocks of Chinese characters. Each character is constructed from a specific sequence and number of strokes. The types of strokes are critical because the order in which they are written affects the overall structure and legibility of the character.

**Basic stroke types include:**

* **Horizontal (横, héng):** A straight line written from left to right.
* **Vertical (竖, shù):** A straight line written from top to bottom.
* **Left-Falling (撇, piě):** A diagonal line falling from top right to bottom left.
* **Right-Falling (捺, nà):** A diagonal line falling from top left to bottom right.
* **Dot (点, diǎn):** A small dot, often written at the beginning or end of a stroke.
* **Rising (提, tí):** A short stroke rising from bottom left to top right.
* **Hook (钩, gōu):** A stroke with a small hook at the end, often added to other strokes.
* **Bend (折, zhé):** A stroke that changes direction sharply, combining horizontal and vertical elements.

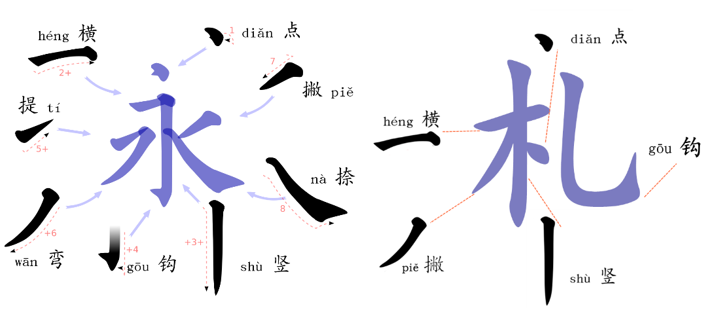


Figure 2.4: Basic Strokes

The order of strokes is essential in Chinese character writing. Characters are written according to specific rules, such as:

1. **Top to bottom**
2. **Left to right**
3. **Horizontal before vertical**
4. **Outside before inside**
5. **Center before sides**

Understanding stroke order is fundamental for both handwriting and input method design, as it influences the way characters are processed and recognized in IMEs.

## ****3. Relationship Between Chinese Character Components****

### ****3.1 Stroke–Radical Relationship****

The relationship between strokes and radicals is essential to understanding the structure of Chinese characters. Each radical is made up of one or more strokes, which must be written in a specific order to form a correct character. For example, the radical **木 (mù, wood)** consists of four strokes: horizontal (横), vertical (竖), left-falling (撇), and right-falling (捺).

### ****3.2 Component Interactions****

Chinese characters are often formed by combining radicals and phonetic components. Some characters are **phono-semantic compounds**, where the radical gives the meaning and the phonetic component provides the pronunciation. For example, **妈 (mā, mother)** combines the radical **女 (nǚ, woman)** with **马 (mǎ, horse)**, where the radical provides the meaning and the phonetic component helps with pronunciation.

The structure and balance of the components within a character are also important. Some characters have a **left-right structure**, while others have a **top-bottom** or **enclosing** structure, which affects the visual balance and readability of the character.

### ****3.3 Stroke Count and Complexity****

The number of strokes in a Chinese character often correlates with its complexity. Simple characters may have only 1 to 5 strokes, such as **一 (yī, one)** or **口 (kǒu, mouth)**. More complex characters may have 6 to 10 strokes, such as **明 (míng, bright)**, and the most complex characters can have more than 20 strokes, such as **龠 (yuè)**.

**Table 3.3:** Radicals and Stroke Numbers

|  |  |  |  |
| --- | --- | --- | --- |
| Radical | English | Pinyin | Stroke count |
| 一 | one | yī | 1 |
| 人 | person | rén | 2 |
| 口 | mouth | kǒu | 3 |
| 心 | heart | xīn | 4 |
| 鸟 | bird | niǎo | 5 |
| 竹 | bamboo | zhú | 6 |
| 豆 | bean | dòu | 7 |

# 4. Design of Chinese input method

### ****4.1 Design Overview****

The Chinese input method designed in this project aims to provide a user-friendly and efficient way for entering Chinese characters using ECCode, a component-based encoding system. The system leverages **Python** and the **Tkinter** library to create a simple and intuitive graphical user interface (GUI), which allows users to input Chinese characters by typing the corresponding ECCode. This input method also supports features like live candidate suggestion, a committed history strip, reverse lookup, and text export functionalities.

The design of the input method focuses on simplicity and usability. It is optimized for users who are familiar with ECCode, providing them with an efficient way to input characters based on predetermined codes. Additionally, the system ensures that all users, regardless of their proficiency in ECCode, can easily type Chinese characters and access relevant characters with ease.

### ****4.2 Dataset****

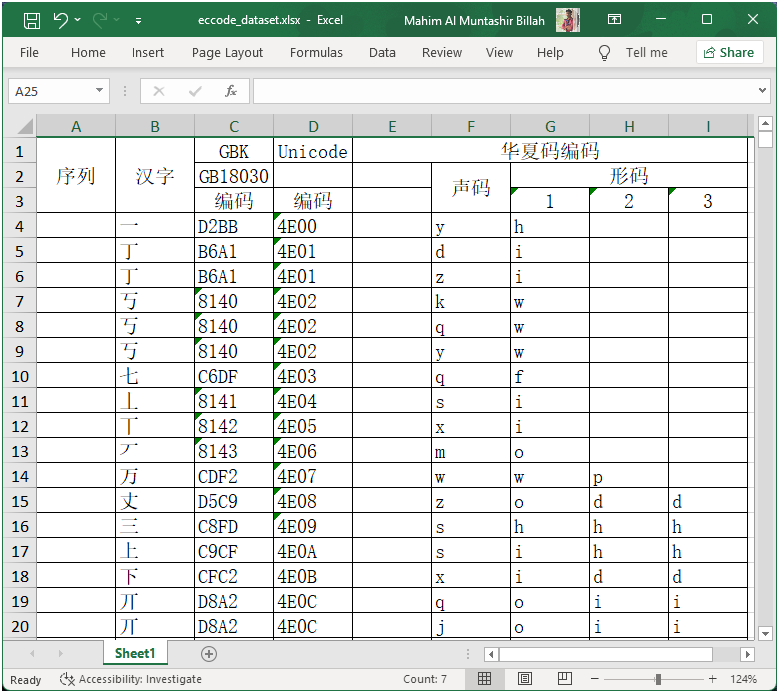
The dataset used for this project is in the form of an Excel file that consists of **1,160 Chinese characters**, each mapped to specific ECCode combinations. These ECCode combinations serve as the key inputs for each character. The dataset includes the following:

* **Character**: The Chinese character itself.
* **ECCode**: The unique code(s) corresponding to the character.

For example:

* Combination **"yh"** corresponds to the character **一 (yī, one)**.
* Combination **"zkii"** corresponds to the character **中 (zhōng, middle)**.

This dataset is read by the Python program, which then maps the ECCode combinations to their respective Chinese characters.



**Figure 4.1**: Sample Dataset with ECCode Combinations

### ****4.3 System Architecture****

The system consists of several components that work together to achieve efficient character input:

1. **Data Loader**: The dataset is read from an Excel file using the pandas library. This component converts the dataset into a form that can be processed by the application, mapping ECCode combinations to their corresponding Chinese characters.
2. **Candidate Generation**: As the user types an ECCode, the program matches the input against the dataset and generates a list of candidate characters that start with the same code. This list is displayed in the GUI for the user to select from.
3. **Commitment and History Management**: Once the user selects a character from the candidate list, it is added to the composed text area. The selected character is also displayed in the **committed history strip** above the code input field. This provides the user with immediate feedback and allows them to track previously selected characters.
4. **Reverse Lookup**: The system includes a feature that allows the user to look up all ECCode combinations associated with a given Chinese character. This feature helps users quickly find alternative ways of typing a character.

**Figure 4.2**: Flowchart of System Architecture

### ****4.4 Design Objectives****

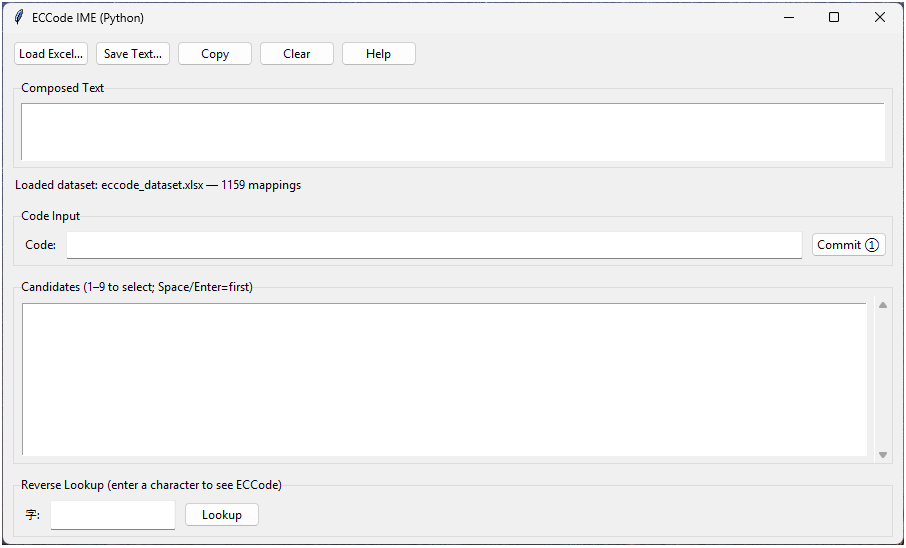
The main design objectives for this input method are as follows:

* **User-friendly Interface**: The GUI should be simple to navigate, with clear sections for input, candidate selection, and history management.
* **Efficiency**: The system should be able to generate candidate characters in real-time as the user types.
* **Accuracy**: The system must return the correct character for a given ECCode input.
* **Transparency**: The relationship between ECCode inputs and Chinese characters should be clear to users, helping them understand how the system works.
* **Error Handling**: The system should handle cases where no matching character is found, displaying an appropriate message to the user.

### ****4.5 Features and User Interaction****

The key features of the input method include:

* **Real-time ECCode Matching**: As the user types, the system provides suggestions of matching Chinese characters based on the ECCode input.
* **Committed History Strip**: A section at the top of the screen where users can see all the characters they have committed so far, making it easier to compose longer pieces of text.
* **Reverse Lookup**: Users can search for ECCode combinations for any given Chinese character, which is useful for learning and error-checking.
* **Text Export**: Users can save their composed text to a file or copy it to the clipboard for use elsewhere.

****

**Figure 4.3**: User Interface of the Chinese Input Method

## ****5. Implementation of Chinese Input Method****

### ****5.1 Implementation Details****

The implementation of the ECCode-based Chinese input method is done entirely in **Python**, utilizing the **Tkinter** library for the graphical user interface and the **pandas** library for handling the dataset.

* **Tkinter** provides a simple way to create the GUI, offering widgets like buttons, text entry fields, labels, and listboxes. The system allows users to type ECCode, view candidates in a list, select characters, and commit them to the text area.
* **Pandas** and **Openpyxl** are used to read and process the dataset from the Excel file. The dataset is loaded into memory, and the ECCode combinations are mapped to their respective Chinese characters, enabling the program to suggest matching characters as the user types.

**Figure 5.1**: Main Window with Code Input and Candidate List

### ****5.2 System Workflow****

The workflow of the system is as follows:

1. **Load Dataset**: The user can load the ECCode dataset using the **"Browse"** button. Once the dataset is loaded, the system is ready to process ECCode input.
2. **Input ECCode**: The user types an ECCode (such as "yh" or "zkii") into the input field. The program immediately searches the dataset for matching characters and displays them in the candidate list.
3. **Select Character**: The user can select a character by clicking on it in the list or by pressing a number key (1–9) corresponding to the candidate's position.
4. **Commit Character**: Once a character is selected, it is committed to the **Composed Text Area**. The character is also added to the **Committed History Strip** above the input field, showing the user the sequence of previously entered characters.
5. **Reverse Lookup**: If the user is unsure of the ECCode for a particular character, they can use the reverse lookup feature to find all possible ECCode combinations for that character.

**Figure 5.2**: Inserting Dataset and Selecting Characters

### ****5.3 Real-Time Accuracy and Feedback****

The system provides immediate feedback by displaying the available characters that match the current input. The real-time nature of the system ensures that users can type quickly and efficiently without unnecessary delays.

For example:

* Typing **"yh"** immediately shows the character **一 (yī)**.
* Typing **"zkii"** shows the character **中 (zhōng)**.

If the user types a random combination that doesn't match any character, the system displays a message such as **“No character associated with key combination”**, ensuring that the user is aware of the error.

**Figure 5.3**: Composed Text with Committed Characters and Error Message

### ****5.4 Performance****

The system runs efficiently with minimal delays, even with a dataset of 1,160 characters. The candidate suggestions are returned in real-time as the user types, and the entire system is responsive, with no noticeable lag during typical usage. The program has been optimized for accuracy and performance, ensuring smooth operation on most modern devices.

### ****5.5 Error Handling and User Assistance****

The program handles errors gracefully, providing feedback when an invalid ECCode is entered. It also helps users correct mistakes by suggesting alternative ECCode combinations using the **Reverse Lookup** function. In cases where no match is found, the system ensures that the user is notified with a clear error message.

**Figure 5.4**: Error Message Display for Invalid ECCode

If you are a Chinese character input method designer, what factors do you think should be considered?

Designing a Chinese character input method requires addressing a variety of factors that influence usability, efficiency, accuracy, and user satisfaction. These factors span across the **linguistic, technical**, and **user experience** domains, and each plays a crucial role in ensuring that the input method is effective, intuitive, and adaptable to different user needs.

#### **Character Coverage and Encoding Standards:**

One of the primary considerations when designing a Chinese input method is ensuring that it covers the full range of characters needed by users. This includes both Simplified and Traditional Chinese characters, as well as characters used in specific regions or ethnic minority languages. The input method should be based on a robust **character encoding** system, such as **Unicode**, to ensure compatibility across different platforms and software. Additionally, the method must support efficient switching between Simplified and Traditional characters to cater to the diverse needs of users in different regions, such as Mainland China, Taiwan, and Hong Kong.

#### **Input Methods and Mode Flexibility:**

Chinese characters can be inputted using various methods, including phonetic-based systems like **Pinyin**, shape-based systems like **ECCode** or **Wubi**, and stroke-based systems. Each of these methods has its advantages and challenges. For instance, phonetic input methods are intuitive for those familiar with the sound of the language but may struggle with homophones. In contrast, shape-based systems like ECCode offer deterministic mapping but require a steeper learning curve. A well-designed input method should support multiple modes and allow users to switch seamlessly between them based on their preference or context. Offering flexibility in input methods can greatly enhance the user experience by catering to a broader range of user proficiencies and needs.

#### **Efficiency and Real-Time Performance:**

Another crucial factor in input method design is **efficiency**. The system should enable users to type characters quickly and accurately, with minimal delays. The method should incorporate features like **real-time candidate suggestions** based on the user's input, **autocomplete** capabilities, and **predictive text** that anticipates the next characters or words. This would help speed up the typing process and improve user productivity, particularly when composing long sentences or complex text. The system must also ensure **low latency** in character selection and display, ideally providing feedback within **50 to 100 milliseconds** after the user inputs a code.

#### **Accuracy and Error Handling:**

Chinese characters are highly context-dependent, and many characters share similar phonetic or shape features. As a result, **accuracy** is a critical concern when designing an input method. The system must provide **precise character suggestions** that match the user’s input. In cases where multiple candidates match the input (e.g., homophones or characters with similar components), the input method should rank the candidates based on **frequency**, **context**, or **recency**, allowing the most likely options to be presented first. Additionally, the method must include effective **error handling** capabilities to tolerate small input mistakes, such as **tone errors** in Pinyin or **incorrect stroke order** in stroke-based methods. The system should be capable of suggesting corrections and offering **fuzzy matching** to handle these discrepancies.

#### **User Interface and Interaction Design:**

The **user interface (UI)** plays a significant role in the overall success of an input method. It must be intuitive, responsive, and easily navigable, particularly for users who may be new to the input method. A good UI ensures that users can easily enter their input, select candidates, and commit the selected characters to the text field. Features such as a **candidate list** with clearly numbered options (e.g., 1-9) and an **easy-to-navigate history strip** where committed characters are shown, are essential for improving efficiency and usability. Additionally, the UI should provide **clear feedback** to the user, such as showing the available candidates as the user types, indicating when no matches are found, and offering suggestions for corrections. Moreover, accessibility features such as **larger fonts**, **high contrast modes**, and **voice-over support** should be considered for users with different needs.

#### **Personalization and Adaptability:**

An effective input method should also support **personalization** to enhance the user experience. This can include allowing users to add custom words or names to a **user dictionary**, enabling faster typing for commonly used terms. The input method should adapt to the user’s typing habits by learning from their previous inputs and adjusting the ranking of candidates accordingly. For example, if a user frequently types a particular name or technical term, the input method should prioritize that term in the candidate list to speed up typing.

#### **Platform Integration and Cross-Platform Compatibility:**

The input method must be designed to work seamlessly across different platforms and devices, including **desktop computers**, **mobile devices**, and **tablets**. This means ensuring compatibility with major operating systems such as **Windows**, **macOS**, and **Linux**, as well as ensuring smooth operation on **Android** and **iOS** devices. The input method should integrate smoothly with other software, such as **word processors**, **web browsers**, and **email clients**, allowing users to input characters without switching between applications. Furthermore, compatibility with other **language input systems** and the ability to switch easily between languages is also essential in a multilingual environment.

#### **Security and Privacy:**

As input methods increasingly rely on cloud-based systems for features like **predictive text** and **user history**, it is essential to address **security** and **privacy concerns**. The input method must ensure that user data is kept private and secure. Ideally, the system should operate **offline** by default to protect user privacy. When cloud services are used, they should be optional, and data should be anonymized and encrypted to prevent unauthorized access. Clear user consent should be obtained before any data is uploaded to the cloud, and users should be able to manage their data preferences, including opting out of cloud-based features.

#### **Cross-Cultural and Multilingual Support:**

Given that Chinese is spoken and written across different regions with variations in character usage (e.g., Simplified versus Traditional Chinese), the input method should support **regional variations** and **multiple dialects**. The ability to switch between **Simplified and Traditional Chinese** is particularly important for users in different Chinese-speaking regions (Mainland China, Taiwan, Hong Kong). The input method should also support **cross-cultural and multilingual typing**, enabling users to input both Chinese and other languages (e.g., English, Japanese, etc.) seamlessly within the same document or conversation.

In conclusion, when designing a Chinese character input method, it is essential to consider factors such as **coverage of characters**, **input methods**, **efficiency**, **accuracy**, **user interface design**, **personalization**, **platform integration**, **security**, and **cross-cultural support**. These elements must work together to create an intuitive, efficient, and adaptable input system that meets the diverse needs of users, whether they are beginners or experts. By focusing on these key aspects, designers can create input methods that enhance typing speed, accuracy, and overall user satisfaction, ultimately enabling seamless communication in Chinese across digital platforms.