2.1入力とASCIIコード

Scanf関数

（要使用scanf函数，和printf一样，需要標準出入力ライブラリstdio.h）

Stdio ：【std】对应标准 【i】对应in 【o】对应out

Scanf函数使用方法

引数、関数、戻り値の概念は、kamata1.1游戏循环介绍里有解释，这里不再赘述

Scanf\_s(“%d”,&num1)

\_s是secure的缩写，安全型scanf

使用时需要输入两个引数 %d : フォーマット指定子

＆num1 : アドレス指定子

ASCIIコード

アルファベットなどを標準的な文字コードのことをASCIIコードと呼ぶ

电脑萤幕画面

AI 生成的内容可能不正确。

char moji = 65;

printf("%c", moji); 打印65

getchar関数

（需要標準出入力ライブラリstdio.h）

入力したうちの１文字目を戻り値として返す関数

Char str;

Str = getchar(); ------如果输入abc，则str的值是a

**清理バッファ的必要性**

**scanf**读取指定类型的数据（如 **%c**、**%d**）后，**不会自动清空バッファ**。

scanf("%c", &ch); // 用户输入 "a\n" 后，バッファ内残留 '**\n**'，**影响下一次读取。**

**什么是バッファ**

用户输入（如键盘）的内容会先存入**バッファ**，**scanf** 和 **getchar** 等函数**从バッファ读取数据，而非直接读取用户输入。**

バッファ**可能残留字符**（如换行符 **\n**、多余的空格等），影响后续输入逻辑。

**如何清理**：

更新：也可直接while (getchar() != '\n'); // 清空输入缓冲区

通过循环***while ((*** ***inputBuffer = getchar()) != '\n')***逐个读取并丢弃バッフsァ内的字符，直到遇到 **\n**，确保バッファ干净。

使用一个变量 ***（inputBuffer）***，像一个临时垃圾桶，它存储每次 **getchar()** 读取的字符，循环里他的值会被**不断覆盖**。

（由 **!= \n** 的条件式）

循环终止时，它的值是 **'\n'**，而**最后一次覆盖的有效字符**是用户**输入回车前的最后一个字符**（如输入 **abc\n**，**inputBuffer** 最终为 **'c'**）。*最后一次条件式得到的inputBuffer不满足条件所以循环停止但他的值还是\n，想利用\n前面那个有效。吗值，再定义一个数来保存最后一次满足条件也就是\n前最后一个字符即可*

**是否利用该值**：

若需保留最后一个字符（如你的代码中赋值给 **searchLetter**），可直接使用；

若仅需清空缓冲区，则无需保存（直接丢弃）。