“加密与解密”——探秘恺撒密码

Python程序综合案例

一、加密

【活动1】 古老的“隐身术”

时间	加密方式
683年	拆字法。将明文中的文字进行组合生成新的字，即为密文，比如“十二月”合起来为“青”
北宋	代码法。北宋进士曾公亮曾搜集了40个常用军事短语，然后对其进行顺序编码：一、请弓;二、请箭;三、请刀;四、请甲;五、请枪旗;六、请锅幕;七、请马;八、请衣赐;九、请粮料……四十、战小胜。军队出征前，指挥机关将用上述短语编码的密码本发给将领，并约定用一首不含重复文字的40字五言律诗与密码相对应。
公元前五世纪	移位法。希罗多德（Herodotus）的《历史》中记载了公元前五世纪，希腊城邦和波斯帝国发生多次冲突和战争。这些战争中希腊城邦中广泛使用了移位法进行加密处理战争通讯信息，使波斯帝国难以获得希腊城邦的军事情报，也就无法提前做军事部署。希腊城邦用来传输军事信息、命令的每段文字都有固定的字数，接密者手中会有一份文字移位说明。解密者拿到密文后，根据文字移位说明进行解密，从而破解其中的军事命令或消息。
古罗马时期	《高卢战记》有描述恺撒曾经使用密码来传递信息，即所谓的“恺撒密码”，它是一种替代密码，通过将字母按顺序推后起3位起到加密作用，如将字母A换作字母D，将字母B换作字母E。因据说恺撒是率先使用加密函的古代将领之一，因此这种加密方法被称为恺撒密码。
第二次世界大战	密码机。在第二次世界大战期间，德国军方启用“恩尼格玛”密码机，密码学在战争中起着非常重要的作用。

一、加密

【悟】加密与解密

加密就是将原始信息（数据）隐匿起来，使之在缺少特殊信息（数据）时不可读。

原始信息（数据）称为明文，加密后的信息（数据）称为密文。

解密将密文还原成明文的过程，又称为解码。

二、恺撒密码的加密算法

【活动2 】揭秘“隐身术”

恺撒在征服高卢、袭击日耳曼和不列颠的多次战斗中频繁使用加密技术。苏托尼厄斯在公元2世纪写的《恺撒传》中对恺撒用过的一种加密技术进行了详细的介绍。恺撒只是简单地将明文中的每一个字母用字母表中该字母后的第3个字母替换。例如，将明文中的a 用d替换，b用e替换，……，z用c替换，这就是恺撒密码。

ASSCII码

ASCII（American Standard Code for Information Interchange，美国信息交换标准代码）

十进制	字符	十进制	字符	十进制	字符	十进制	字符	十进制	字符	十进制	字符	十进制	字符	十进制	字符	十进制	字符	十进制	字符	十进制	字符	十进制	字符	十进制	字符	十进制	字符	十进制	字符
0		16	▶	32		48	0	64	@	80	P	96	`	112	p	128	Ç	144	É	160	á	176	▒	192	└	208	╨	224	α	240	≡
1	☺	17	◀	33	!	49	1	65	A	81	Q	97	a	113	q	129	ü	145	æ	161	í	177	▓	193	┴	209	╤	225	ß	241	±
2	☻	18	↕	34	"	50	2	66	B	82	R	98	b	114	r	130	é	146	Æ	162	ó	178	▓	194	┬	210	╥	226	Γ	242	≥
3	♥	19	‼	35	#	51	3	67	C	83	S	99	c	115	s	131	â	147	ô	163	ú	179	▓	195	├	211	╙	227	π	243	≤
4	♦	20	¶	36	$	52	4	68	D	84	T	100	d	116	t	132	ä	148	ö	164	ñ	180	▓	196	─	212	╘	228	Σ	244	⌠
5	♣	21	§	37	%	53	5	69	E	85	U	101	e	117	u	133	à	149	ò	165	Ñ	181	▓	197	┼	213	╒	229	σ	245	⌡
6	♠	22	▬	38	&	54	6	70	F	86	V	102	f	118	v	134	å	150	û	166	ª	182	▓	198	╞	214	╓	230	µ	246	÷
7	•	23	↨	39	'	55	7	71	G	87	W	103	g	119	w	135	ç	151	ù	167	º	183	▓	199	╟	215	╫	231	τ	247	≈
8	◘	24	↑	40	(	56	8	72	H	88	X	104	h	120	x	136	ê	152	ÿ	168	¿	184	▓	200	╚	216	╪	232	Φ	248	°
9	○	25	↓	41	)	57	9	73	I	89	Y	105	i	121	y	137	ë	153	Ö	169	⌐	185	▓	201	╔	217	┘	233	Θ	249	•
10	◙	26	→	42	*	58	:	74	J	90	Z	106	j	122	z	138	è	154	Ü	170	¬	186	▓	202	╩	218	┌	234	Ω	250	·
11	♂	27	←	43	+	59	;	75	K	91	[	107	k	123	{	139	ï	155	¢	171	½	187	▓	203	╦	219	▓	235	δ	251	√
12	♀	28	∟	44	,	60	<	76	L	92	\	108	l	124	|	140	î	156	£	172	¼	188	▓	204	╠	220	▄	236	∞	252	n
13	♪	29	↔	45	-	61	=	77	M	93	]	109	m	125	}	141	ì	157	¥	173	¡	189	▓	205	═	221	▌	237	φ	253	²
14	♫	30	▲	46	.	62	>	78	N	94	^	110	n	126	~	142	Ä	158	₧	174	«	190	▓	206	╬	222	▐	238	∈	254	■
15	☼	31	▼	47	/	63	?	79	O	95	_	111	o	127	⌂	143	Å	159	ƒ	175	»	191	▓	207	╧	223	▀	239	∩	255	ÿ

常用的字符串函数

函数名	函数功能	应用举例
len（s）	求字符串s的长度	len(“china”)=5
isupper(s)	判断字符s是否为大写	isupper(‘A’)返回值为True
islower(s)	判断字符s是否为小写	islower(’Z’)返回值为False
ord(s)	返回字符s的ASCII码值	ord(‘A’)=65
chr(num)	返回num对应的字符	chr（65）=A

Python中遍历字符串的方法

- 1.下标的方法，索引从0开始，结合len()函数使用
- 2.for in的方法

- **字符串连接“+”**

探究一：恺撒密码的加密算法（一）

算法分析： 将明文中的每一个字母用字母表中该字母后的第3个字母替换。例如，将明文中的a 用d替换，b用e替换，……，z用c替换，这就是恺撒密码。

抽象建模：分类讨论，得出公式

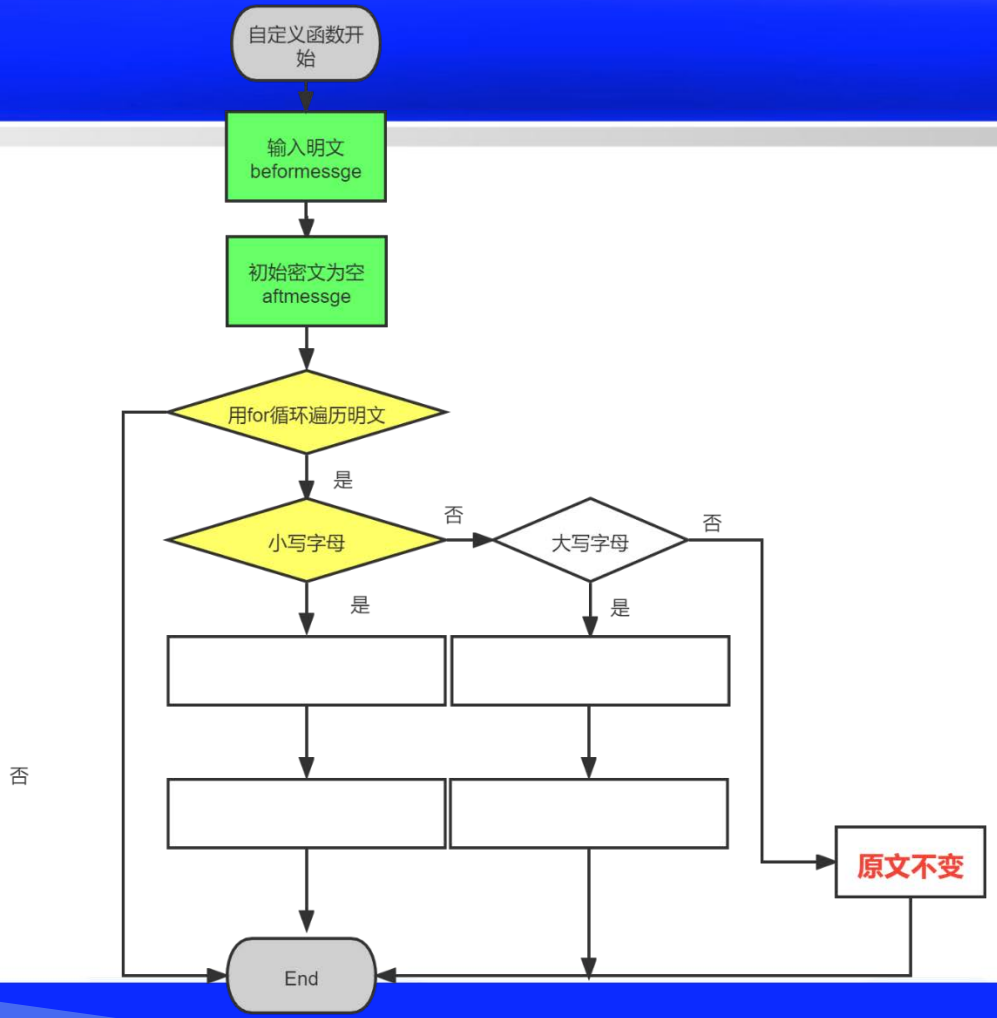
（1）小写字母‘a’<=c[i]<=‘z 公式：**chr(ord(c[i])+3)**

（2）大写字母‘A’<=c[i]<=‘Z’公式： **chr(ord(c[i])-23)**

（3）其他情况：保持不变

明文C	a~z	A~Z	其他
密文b			不变
规律	用小写字母a的ASCII码与当前字符ASCII码与字符a的ASCII码的差值加3，与26取余	用小写字母A的ASCII码与当前字符ASCII码与字符A的ASCII码的差值加3，与26取余	不变
公式	code=ord('a')+(ord(char) - ord('a') + 3) % 26 b[i]=char(code)	Code=ord('A')+(ord(char)-ord('A')+3) % 26 b[i]=char(code)	b[i]=c[i]

算法一的流程图


自定义函数开始
输入明文
beformessge
初始密文为空
aftmessge
用for循环遍历明文
是
小写字母
否
大写字母
否
是
是
否
原文不变
End

恺撒密码的加密算法

恺撒密码的加密算法核心代码

```
#定义加密函数，对字母进行加密，即向后移动key位，其他字符不加密。
def cipher(befmessage, key):
    aftmessage = ''
    for char in befmessage:
        if char.isupper():    #对大写字母进行加密
            code = ord('A')+(ord(char)-ord('A')+key) % 26
            aftmessage = aftmessage+chr(code)
        elif char.islower():  #对小写字母进行加密
            code =ord('a')+(ord(char) - ord('a') + key) % 26
            aftmessage = aftmessage+chr(code)
        else:
            aftmessage = aftmessage+char#字母以外的其他字符不进行加密
    return aftmessage
```

探究二：恺撒密码的加密算法（二）

算法分析： 将明文中的每一个字母用字母表中该字母后的第3个字母替换。例如，将明文中的a 用d替换，b用e替换，……，z用c替换，这就是恺撒密码。

明文c	a~w	A~W	x~z	X~Z	其他
密文b	d~z	D~Z	a~c	A~C	不变
规律	ASCII码+3	ASCII码+3	ASCII码+3-26	ASCII码+3-26	不变
公式	'a'<=c[i]<='w' or 'A'<=c[i]<='W'		'x'<=c[i]<='z' or 'X'<=c[i]<='Z'		
	chr(ord(c[i])+3)		chr(ord(c[i])-23)		c[i]

chr(ord(c[i])+3)　　chr(ord(c[i])-23)　　'x'<=c[i]<='z' or 'X'<=c[i]<='Z'

探究二：恺撒密码的加密算法（二）

算法分析： 将明文中的每一个字母用字母表中该字母后的第3个字母替换。例如，将明文中的a 用d替换，b用e替换，……，z用c替换，这就是恺撒密码。

抽象建模：分类讨论，得出公式

（1） ‘a’ <=c[i]<= ‘w’ or ‘A’ <=c[i]<= ‘W’ 公式：**chr(ord(c[i])+3)**

（2）'x'<=c[i]<='z' or 'X'<=c[i]<= ‘Z’ 公式：**chr(ord(c[i])-23)**

（3）其他情况：保持不变

明文c	a~w	A~W	x~z	X~Z	其他
密文b	d~z	D~Z	a~c	A~C	不变
规律	ASCII码+3	ASCII码+3	ASCII码+3-26	ASCII码+3-26	不变
公式	'a'<=c[i]<='w' or 'A'<=c[i]<='W'		'x'<=c[i]<='z' or 'X'<=c[i]<='Z'		
	chr(ord(c[i])+3)		chr(ord(c[i])-23)		c[i]

算法二的流程图

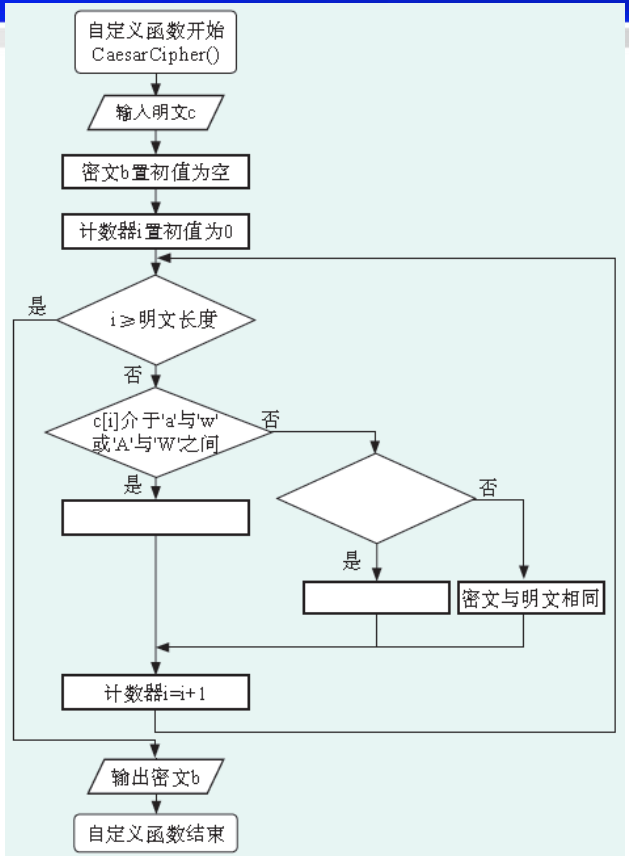


图3.4.1　恺撒加密算法流程图

探究二：恺撒密码的加密算法

- 代码实现

```
mingwen=input("请输入明文：")
secret=""
n=0
while n<=len(mingwen)-1:
    if 'a'<=mingwen[n]<='w' or 'A'<=mingwen[n]<='W':
        secret=secret+chr(ord(mingwen[n])+3)
    elif 'x'<=mingwen[n]<='z' or 'X'<=mingwen[n]<='Z':
        secret=secret+chr(ord(mingwen[n])-23)
    else:
        secret=secret+mingwen[n]
    n=n+1
print(secret)
print("运行完毕，请按回车键退出……")
```

【拓展】：恺撒密码的解密算法

【想一想】恺撒密码的解密算法核心代码

三、恺撒密码的解密算法

恺撒密码的解密算法核心代码

```
c=input(“请输入密文：”)
b= ‘’
for i in range(len(c)):                    #获取密文内容的每一个字符，并解密
    if 'd'<=c[i]<='z' or 'D'<=c[i]<='Z':    #判断d～z或D～Z间的字母
        b=b+chr(ord(c[i])-3)                #生成明文
    elif 'a'<=c[i]<='c' or 'A'<=c[i]<='C': #判断a～c或A～C间的字母
        b=b+chr(ord(c[i])+23)               #生成明文
    else:
        b=b+c[i]                            #字母以外的密文不变
print(b)
```

四、项目小结

本内容学科核心素养之三层架构

学科知识层：数据类型（字符串、字符），字符与ASCII码，函数ord()与chr(),循环语句，分支语句，条件（逻辑）表达式

问题解决层：算法（加密与解密都是对字符的替换），测试

学科思维层：抽象（恺撒密码抽象为字符串的替换操作）、建模（把明文、密文当作字符串处理，字符串就是一种模型）