

COMPETENCIA INTERNA DE PROGRAMACIÓN

Problema WERTYU

Ing. Marina Cardenas

**UNIVERSIDAD TECNOLÓGICA NACIONAL
FACULTAD REGIONAL CÓRDOBA
DEPARTAMENTO DE INGENIERÍA EN SISTEMAS DE INFORMACIÓN
MSLABS - LABORATORIO DE INVESTIGACIÓN DE SOFTWARE**

Problema WERTYU

Enunciado

Un error de tipeo común, es colocar sus manos sobre el teclado una posición a la derecha de la posición correcta. Entonces "Q" es pulsada como "W", "J" es pulsada como "K" y así con el resto de las teclas (en verde en la figura). Su tarea es decodificar un mensaje escrito de esta manera.

Problema WERTYU

Entrada

La entrada consistirá de diversas líneas de texto. Cada línea puede contener dígitos, espacios, letras en mayúsculas (excepto "Q", "A", "Z") o símbolos de puntuación (excepto el acento " ` "). Las teclas con nombre (Tab, BackSp, Control, Alt, Enter) no serán representadas en la entrada.

Salida

Usted debe reemplazar cada símbolo por el inmediato anterior a la izquierda sobre el teclado de la figura. El espacio en blanco en la entrada debe ser copiado en la salida.

Problema WERTYU

Ejemplo

Entrada

RDYS RD IMS [TIRNS/

Salida

ESTA ES UNA PRUEBA.

Teclado QWERTY a utilizar

	1	2	3	4	5	6	7	8	9	0	-	=	Backup	
<u>Tab</u>	Q	W	E	R	T	Y	U	I	O	P	[]	\	
<u>A</u>	S	D	F	G	H	J	K	L	;	'	<u>Enter</u>			
<u>Z</u>	X	C	V	B	N	M	,	.	/					
<u>Control</u>		<u>Alt</u>						<u>Alt</u>	<u>Control</u>					



Gris	Estos símbolos no se presentarán en el texto de entrada
Celeste	El espacio en blanco en el texto de entrada se traduce como espacio en blanco en el texto de salida
Verde	Estos símbolos cuando aparezcan en el texto de entrada, deben traducirse en el texto de salida, por el símbolo que aparece inmediatamente a la izquierda

Problema WERTYU

Análisis del Problema



El programa solicitado es un *programa traductor* regido por una función que traduce cada símbolo en su entrada, por el símbolo ubicado inmediatamente a la izquierda en el teclado mostrado en la figura, en su salida, salvo el espacio en blanco que se traduce sin cambios.

Hay algunos problemas de especificación en el enunciado, que deben discutirse:

Problema WERTYU

Formalmente el problema se puede describir como una **Máquina de Mealy**:

$$Me = (E, S, Q, q, f, g)$$

donde:

E= alfabeto de símbolos de entrada

= {1,2,3,4,5,6,7,8,9,0,-,=,W,E,R,T,Y,U,I,O,P,[,],\,S,D,F,G,H,J,K,L,;,',X,C,V,B,N,M,.,./,b}

S= alfabeto de símbolos de salida

= {'1,2,3,4,5,6,7,8,9,0,-,Q,W,E,R,T,Y,U,I,O,P,[,],A,S,D,F,G,H,J,K,L,;Z,X,C,V,B,N,M,.,./,b}

Q= conjunto finito de estados

= {q}

q = estado inicial

f = función de transición de estado a estado, $f: Q \times E \rightarrow Q$

f	1	2	3	4	5	6	7	8	9	0	-	=	W	E	R	T	Y	U	I	O	P	[]	\	S	D	F	G	H	J	K	L	;	'	X	C	V	B	N	M	,	.	/				
q	q	q	q	q	q	q	q	q	q	q	q	q	q	q	q	q	q	q	q	q	q	q	q	q	q	q	q	q	q	q	q	q	q	q	q	q	q	q	q	q	q	q	q	q	q	q	q


g = función de salida, $g: Q \times E \rightarrow S$

g	1	2	3	4	5	6	7	8	9	0	-	=	W	E	R	T	Y	U	I	O	P	[]	\	S	D	F	G	H	J	K	L	;	'	X	C	V	B	N	M	,	.	/	
q	`	1	2	3	4	5	6	7	8	9	0	-	Q	W	E	R	T	Y	U	I	O	P	[]	A	S	D	F	G	H	J	K	L	;	Z	X	C	V	B	N	M	,	.	


Problema WERTYU

El funcionamiento de ésta máquina es, resumidamente, como sigue:
Leer símbolos en la entrada y, por cada uno de ellos, emitir en la salida g(q, símbolo leído).

Algorítmicamente:



```
#include <stdio.h>
#include <string.h>
#include <conio.h>
int main()
{
    char cad[300];
    int i=0;
    char c;
    gets(cad);
    while(strcmp(cad,"")!=0){
        i=0;
        while(cad[i]!='\0'){
            c=Reemplazo(cad[i]);
            putchar(c);
            i++;
        }
        gets(cad);
    }
    return 1;
}
```



```
import java.util.Scanner;

public class Werty {
    public static void main( String args[] )
    {
        String cad;
        int i=0;
        char c,s;
        Scanner sc=new Scanner(System.in);
        cad=sc.nextLine();
        String aux="";
        while(cad.compareTo("")!=0){
            i=0;
            while(i<cad.length()){
                s=Reemplazo(cad.charAt(i));
                aux+=s;
                i++;
            }
            System.out.println(aux);
            aux="";
            cad=sc.nextLine();
        }
    }
}
```

Problema WERTYU



```
char Reemplazo (char entrada)
{
    char salida;
    switch(entrada)
    {
        case '1': salida = '`'; break;
        case '2': salida = '1'; break;
        case '3': salida = '2'; break;
        case '4': salida = '3'; break;
        case '5': salida = '4'; break;
        case '6': salida = '5'; break;
        case '7': salida = '6'; break;
        case '8': salida = '7'; break;
        case '9': salida = '8'; break;
        case '0': salida = '9'; break;
        case '-': salida = '0'; break;
        case '=': salida = '-'; break;
        case 'W': salida = 'Q'; break;
        case 'E': salida = 'W'; break;
        case 'R': salida = 'E'; break;
        case 'T': salida = 'R'; break;
        case 'Y': salida = 'T'; break;
        case 'U': salida = 'Y'; break;
        case 'I': salida = 'U'; break;
        case 'O': salida = 'I'; break;
        case 'P': salida = 'O'; break;
        case '[': salida = 'P'; break;
        ...

        case ']': salida = '['; break;
        case '\\': salida = ']'; break;
        case 'S': salida = 'A'; break;
        case 'D': salida = 'S'; break;
        case 'F': salida = 'D'; break;
        case 'G': salida = 'F'; break;
        case 'H': salida = 'G'; break;
        case 'J': salida = 'H'; break;
        case 'K': salida = 'J'; break;
        case 'L': salida = 'K'; break;
        case ';': salida = 'L'; break;
        case '\\': salida = ';'; break;
        case 'X': salida = 'Z'; break;
        case 'C': salida = 'X'; break;
        case 'V': salida = 'C'; break;
        case 'B': salida = 'V'; break;
        case 'N': salida = 'B'; break;
        case 'M': salida = 'N'; break;
        case ',': salida = 'M'; break;
        case '.': salida = ','; break;
        case '/': salida = '.'; break;
        case ' ': salida = ' '; break;
    }
    return salida;
}
```

Problema WERTYU



```
public static char Reemplazo (char entrada)
{
    char salida=' ';
    switch(entrada)
    {
        case '1': salida = '`'; break;
        case '2': salida = '1'; break;
        case '3': salida = '2'; break;
        case '4': salida = '3'; break;
        case '5': salida = '4'; break;
        case '6': salida = '5'; break;
        case '7': salida = '6'; break;
        case '8': salida = '7'; break;
        case '9': salida = '8'; break;
        case '0': salida = '9'; break;
        case '-': salida = '0'; break;
        case '=': salida = '-'; break;
        case 'W': salida = 'Q'; break;
        case 'E': salida = 'W'; break;
        case 'R': salida = 'E'; break;
        case 'T': salida = 'R'; break;
        case 'Y': salida = 'T'; break;
        case 'U': salida = 'Y'; break;
        case 'I': salida = 'U'; break;
        case 'O': salida = 'I'; break;
        case 'P': salida = 'O'; break;
        case '[': salida = 'P'; break;
        ...
        case ']': salida = '['; break;
        case '\\': salida= ']' ; break;
        case 'S': salida = 'A'; break;
        case 'D': salida = 'S'; break;
        case 'F': salida = 'D'; break;
        case 'G': salida = 'F'; break;
        case 'H': salida = 'G'; break;
        case 'J': salida = 'H'; break;
        case 'K': salida = 'J'; break;
        case 'L': salida = 'K'; break;
        case ';': salida = 'L'; break;
        case '\\': salida= ';'; break;
        case 'X': salida = 'Z'; break;
        case 'C': salida = 'X'; break;
        case 'V': salida = 'C'; break;
        case 'B': salida = 'V'; break;
        case 'N': salida = 'B'; break;
        case 'M': salida = 'N'; break;
        case ',': salida = 'M'; break;
        case '.': salida = ','; break;
        case '/': salida = '.'; break;
        case ' ': salida = ' '; break;
    }
    return salida;
}
```

Problema WERTYU

Otra Opción: Armar un vector de 48 elementos y cargarlo con todos los símbolos de $E \cup S$ en el orden especificado en el teclado, para luego implementar la función **g** como una **búsqueda secuencial** en el vector del símbolo leído y, al encontrarlo, retornar el elemento anterior.

C

```
char ES[48] = {'`', '1', '2', ..., '.', '/', ' '};
char Reemplazo (char entrada)
{
    char salida=' ';
    for (int i=1; i < 48; i++)
    {
        if (ES[i] == entrada)
        {
            salida = ES[i-1];
            break;
        }
    }
    return salida;
}
```

JAVA

```
public static char Reemplazo (char entrada)
{
    char ES[]= {'`','1','2',..., '.', '/', ' '};
    char salida=' ';
    for (int i=1; i < 48; i++)
    {
        if (ES[i] == entrada)
        {
            salida = ES[i-1];
            break;
        }
    }
    return salida;
}
```

Problema WERTYU

Otras Alternativas:

- Armar cuatro vectores (uno por línea del teclado) para hacer búsqueda secuencial en el primero si el símbolo de entrada es un número o en los otros tres si el símbolo de entrada es una letra. Si la entrada es un símbolo de puntuación, se puede buscar en los cuatro vectores desde atrás hacia delante sólo en los últimos tres lugares. Si el símbolo de entrada es espacio en blanco, copiarlo como está.

Esta es una alternativa sobre el anterior algoritmo que, con una mayor complejidad, puede acelerar la búsqueda del símbolo de entrada.

- En vez de utilizar vectores, usar cadenas de caracteres (Strings) como almacenamiento de los símbolos del teclado y efectuar la búsqueda usando una función del lenguaje que permita extraer la subcadena de largo uno en la posición i-ésima. Estas funciones en los lenguajes de programación suelen implantarse internamente con vectores, por lo que su uso es una alternativa pero seguramente de menor eficiencia que usar directamente vectores.

- Armar un vector de pares ordenados de símbolos (**entrada, salida**) y ordenarlo por la primera componente para efectuar una **búsqueda binaria** en este vector del símbolo de entrada ingresado, devolviendo la segunda componente al encontrar. (más rápido que la búsqueda secuencial pero ocupa el vector el doble del tamaño y es más complejo).

Problema WERTYU

- Usar un vector de 255 elementos indexado por el código ASCII del símbolo de entrada. (muy rápido por acceso directo al vector, pero ocupa más memoria y requiere tener a mano una tabla ASCII).

0	24	†	48	0	72	H	96		120	x	144	E	168	¿	192	L	216	‡	240	≡
1	25	↓	49	1	73	I	97	a	121	y	145	æ	169	ƒ	193	⊥	217	‡	241	±
2	26	+	50	2	74	J	98	b	122	z	146	æ	170	ŕ	194	⌈	218	‡	242	≥
3	27	+	51	3	75	K	99	c	123	(147	ô	171	ŕ	195	⌈	219	‡	243	≤
4	28	⌈	52	4	76	L	100	d	124)	148	ö	172	ŕ	196	⌈	220	‡	244	∫
5	29	⌈	53	5	77	M	101	e	125)	149	ö	173	ŕ	197	⌈	221	‡	245	∫
6	30	⌈	54	6	78	N	102	f	126	~	150	û	174	ŕ	198	⌈	222	‡	246	÷
7	31	⌈	55	7	79	O	103	g	127	Δ	151	ù	175	ŕ	199	⌈	223	‡	247	≈
8	32	⌈	56	8	80	P	104	h	128	Ç	152	ü	176	ŕ	200	⌈	224	α	248	°
9	33	†	57	9	81	Q	105	i	129	ü	153	ö	177	ŕ	201	⌈	225	β	249	•
10	34	⌈	58	:	82	R	106	j	130	é	154	ü	178	ŕ	202	⌈	226	Γ	250	•
11	35	#	59	:	83	S	107	k	131	ä	155	ç	179	ŕ	203	⌈	227	Π	251	√
12	36	\$	60	<	84	T	108	l	132	ä	156	ç	180	ŕ	204	⌈	228	Σ	252	ⁿ
13	37	%	61	=	85	U	109	m	133	ä	157	ŕ	181	ŕ	205	⌈	229	σ	253	˚
14	38	&	62	>	86	V	110	n	134	ä	158	ŕ	182	ŕ	206	⌈	230	μ	254	■
15	39	•	63	?	87	W	111	o	135	ç	159	ŕ	183	ŕ	207	⌈	231	γ	255	■
16	40	(64	@	88	X	112	p	136	è	160	ä	184	ŕ	208	⌈	232	ø		
17	41)	65	A	89	Y	113	q	137	è	161	í	185	ŕ	209	⌈	233	ø		
18	42	x	66	B	90	Z	114	r	138	è	162	ó	186	ŕ	210	⌈	234	Ω		
19	43	+	67	C	91	[115	s	139	ŕ	163	ú	187	ŕ	211	⌈	235	ø		
20	44	,	68	D	92	\	116	t	140	î	164	ñ	188	ŕ	212	⌈	236	∞		
21	45	-	69	E	93]	117	u	141	î	165	ñ	189	ŕ	213	⌈	237	ø		
22	46	.	70	F	94	^	118	v	142	ä	166	ä	190	ŕ	214	⌈	238	€		
23	47	/	71	G	95	_	119	w	143	ä	167	ø	191	ŕ	215	⌈	239	Ω		

FIN DE LA PRESENTACIÓN

**Presentación basada en
soluciones propuestas por el
Ing. Juan Carlos Vázquez**