

## Practica No 7

**Objetivo general:** Aprender a usar el método de ordenación llamado de la burbuja.

### Desarrollo:

En esta practica realizamos cuatro programas todos relacionados con *el método de la burbuja* el cual es un método de ordenación de arreglos o mejor dicho de cadenas y es llamado así por que sus valores parecen ascender.

El primer programa que realizamos es la ordenación de forma ascendente de una cadena de 1000 elementos los cuales debían ser números para lo cual preferimos hacer uso de un programa que realizamos con anterioridad el cual generaba números aleatorios mediante el comando random.

Entonces a manera de *metodología* lo que usamos es un ciclo for que me daba el índice de dos cadenas que estaban dentro de este y los cuales eran llenados con el valor obtenido mediante el comando randomize. Después de haber llenado las dos cadenas mediante el método de la burbuja ordenábamos una de las cadenas y ya ordenadas iniciamos un ciclo for que va desde el cero hasta la cantidad de elementos menos uno (es el numero de índice de nuevo) dentro de este mandamos a imprimir los valores de la cadena ya ordenada y la que esta desordenada para que se note la diferencia también junto con ello también imprimo el numero de elemento que es (el numero de índice ofrecido por el ciclo for mas uno).

Por lo tanto el código fuente del programa quedo de la siguiente forma:

```
/*
FECHA: 07-NOV-06
PROGRAMA: 22
OBJETIVO: /*COMENZARA ANALIZAR EL FUNCIONAMIENTO DEL METODO DE ORDENACIÓN
DE LA BURBUJA Y VERIFICAR QUE FUNCIONA */
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include <conio.h>
#include <dos.h>
main ()
{
int i,j,u,x=1,l,g=5;
int a[1000],k,pasada,aux,b[1000];
textmode (C4350);
textcolor (RED);
textbackground (CYAN);
clrscr ();
for (l=1;l<=80;l++)
{
gotoxy (l,1);
printf ("*");
delay (g);
}
for (l=1;l<=49;l++)
{
gotoxy(80,l);
printf ("*");
delay (g);
}
}
```

```

for (l=80; l>=1;--l)
{
gotoxy (l,49);
printf ("*");
delay (g);
}
for (l=49; l>=1;--l)
{
gotoxy (1,l);
printf ("*");
delay (g);
}
randomize ();
for (j=0; j<1000; j++)
{
i=random (255); //generamos un numero aleatorio y lo guardamos en i
a[j]=i; //Llenamos una cadena que sera la que ordenaremos
b[j]=i; //Llenamos la cadena que se mostrara desordenada
}
for (pasada=1; pasada<=999; pasada++) // indicamos de veces que ordenaremos todos los elementos
{
for (k=0; k<=998; k++) //vamos ordenando un elemento
{
if (a[k]>a[k+1]) //comparacion
{
aux=a[k];
a[k]=a[k+1];
a[k+1]=aux;
}
}
}
printf ("\n*\t No de dato\t Datos sin ordenar \t Datos ordenados\n\n");
for (k=0; k<1000; k++)
{
u=x*42; //Es usado para mostrar por pantallas los numeros
if (k==u)
{
x++;
gotoxy (5,47);
printf ("Presiona una tecla para continuar");
getch ();
clrscr ();
for (l=1; l<=80; l++)
{
gotoxy (1,l);
printf ("*");
delay (g);
}
for (l=1; l<=49; l++)
{
gotoxy(80,l);
printf ("*");
delay (g);
}
for (l=80; l>=1;--l)
{

```

```

        gotoxy (1,49);
        printf ("*");
        delay (g);
    }
    for (l=49;l>=1;--l)
    {
        gotoxy (1,l);
        printf ("*");
        delay (g);
    }
    printf ("\n*\t No de dato\tDatos sin ordenar \t Datos ordenados");
    printf ("\n\n*\t\t%d\t\t%d\t\t%d\n",k,b[k],a[k]); // imprimimos el numero de elemento y las dos cadenas
    }
else
{
    printf ("*\t\t%d\t\t%d\t\t%d\n",k+1,b[k],a[k]);
}
}
getch ();
}
/*CONCLUSIONES: ESTSE PROGRAMA NOS SIRVIO DE REFERENCIA
PARA EL DESARROLLO DE OTROS MAS QUE SE MOSTRARAN A CONTINUACIÓN
YA QUE EL METODO DE ORDENACIÓN QUE SE USA ES INVARIABLE
TANTO PARA NUMEROS COMO PARA CARACTERES */

```

Dentro del programa explicamos mas o menos como es que se ordenan los elementos en este caso están ordenados de forma ascendente o sea vamos del menor valor al mayor.

La corrida cuanta con demasiadas pantallas por lo que solo imprimire tres verificando que sea correcto:

No de dato	Datos sin ordenar	Datos ordenados
1	95	0
2	243	0
3	152	0
4	131	0
5	128	0
6	81	1
7	120	1
8	34	1
9	203	1
10	173	1
11	40	1
12	226	1
13	71	1
14	159	2
15	55	2
16	37	3
17	101	3
18	149	3
19	167	3
20	234	3
21	57	4
22	223	4
23	10	4
24	107	4
25	35	4
26	48	4
27	34	4
28	233	4
29	52	5
30	194	6
31	170	6
32	68	6
33	54	6
34	243	7
35	84	7
36	58	8
37	166	8
38	113	8
39	146	8
40	214	8
41	125	9
42	198	9

Presiona una tecla para continuar...

```

C:\bc31\TMP\NUMEBUR.EXE
*****
No de dato      Datos sin ordenar      Datos ordenados
*****
504             213                    125
506             198                    125
507             213                    125
508             97                     125
509             67                     125
510             66                     125
511             31                     126
512             234                    126
513             182                    126
514             36                     126
515             173                    127
516             289                    127
517             212                    127
518             298                    128
519             178                    128
520             45                     128
521             77                     128
522             182                    128
523             125                    128
524             178                    129
525             131                    129
526             44                     129
527             224                    129
528             186                    130
529             116                    130
530             212                    130
531             287                    130
532             59                     131
533             43                     131
534             125                    131
535             223                    131
536             189                    131
537             167                    131
538             237                    132
539             35                     133
540             82                     134
541             138                    134
542             86                     134
543             49                     134
544             240                    134
545             87                     135
546             34                     135
*****
Presiona una tecla para continuar
*****

C:\bc31\TMP\NUMEBUR.EXE
*****
No de dato      Datos sin ordenar      Datos ordenados
*****
966             186                    245
968             13                     245
969             191                    246
970             153                    246
971             233                    247
972             161                    247
973             41                     248
974             199                    248
975             98                     248
976             125                    248
977             124                    248
978             193                    249
979             115                    249
980             282                    250
981             72                     250
982             28                     250
983             224                    250
984             28                     250
985             198                    250
986             48                     251
987             113                    251
988             173                    251
989             192                    251
990             31                     252
991             184                    252
992             98                     252
993             137                    253
994             187                    253
995             198                    253
996             76                     254
997             114                    254
998             153                    254
999             164                    254
1000            92                     254
*****

```

El segundo programa consiste en nuevamente ordenar 1000 elementos pero esta vez de forma descendente. La metodología a seguir es la misma y en lo único en que cambia el programa es que ahora en donde tenemos la comparación nos fijaremos si el elemento en cuestión es menor al que le sigue (en la cadena) si se cumple la condición entonces intercambiamos los lugares de ambos elementos.

Por lo cual el código fuente queda de la siguiente forma:

```

/*
FECHA: 07-NOV-06
PROGRAMA: 23
OBJETIVO: /*ANALIZAR LA FORMA EN QUE EL METODO DE LA BURBUJA
SE PUEDE INVERTIR PARA ORDENAR DE FORMA DESCENDENTE LOS NUMEROS GENERADOS
EN LA CADENA */

```

```

#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include <conio.h>
#include <dos.h>
main ()
{
int i,j,u,x=1,l,g=5;
int a[1000],k,pasada,aux,b[1000];
textmode (C4350);
textcolor (RED);
textbackground (CYAN);
clrscr ();
for (l=1;l<=80;l++)
{
gotoxy (l,1);
printf ("*");
delay (g);
}
for (l=1;l<=49;l++)
{
gotoxy(80,l);
printf ("*");
delay (g);
}
for (l=80; l>=1;--l)
{
gotoxy (l,49);
printf ("*");
delay (g);
}
for (l=49;l>=1;--l)
{
gotoxy (1,l);
printf ("*");
delay (g);
}
randomize ();
for (j=0;j<1000;j++)
{
i=random (255);
//printf ("random number in the 0-255 range: %c %d \n\n",i,i);
//delay (2);
a[j]=i;
b[j]=i;
}
for (pasada=1;pasada<=999;pasada++)
{
for (k=0;k<=998;k++)
{
if (a[k]<a[k+1]) //Comparamos si es menor
{
aux=a[k];
a[k]=a[k+1];
a[k+1]=aux;
}
}
}
}

```



```

C:\Abc31\TMP\NUMEBUR2.EXE
*****
No de dato      Datos sin ordenar      Datos ordenados
*****
1              14                    254
2              231                   254
3              65                    254
4              132                   253
5              196                   253
6              74                    253
7              80                    253
8              225                   252
9              5                    252
10             92                    252
11             200                   252
12             213                   252
13             246                   250
14             80                    250
15             336                   250
16             4                    250
17             14                    250
18             250                   250
19             254                   250
20             116                   249
21             72                    249
22             216                   249
23             93                    248
24             181                   248
25             188                   248
26             53                    247
27             113                   247
28             13                    247
29             252                   246
30             50                    246
31             184                   246
32             187                   245
33             45                    245
34             30                    245
35             247                   244
36             177                   244
37             95                    244
38             219                   244
39             71                    244
40             186                   243
41             31                    242
42             285                   242
*****
Presiona una tecla para continuar_
*****

```

```

C:\Abc31\TMP\NUMEBUR2.EXE
*****
No de dato      Datos sin ordenar      Datos ordenados
*****
504             164                   113
506             89                    113
507             17                    112
508             222                   112
509             215                   112
510             49                    112
511             29                    112
512             53                    112
513             150                   111
514             35                    111
515             183                   111
516             51                    111
517             52                    110
518             73                    110
519             189                   110
520             239                   110
521             160                   109
522             15                    109
523             46                    109
524             59                    109
525             204                   109
526             228                   109
527             203                   109
528             26                    108
529             120                   108
530             161                   108
531             179                   108
532             120                   108
533             44                    107
534             39                    107
535             28                    107
536             181                   107
537             141                   107
538             189                   107
539             169                   106
540             195                   106
541             84                    106
542             240                   106
543             174                   106
544             171                   106
545             131                   106
546             181                   185
*****
Presiona una tecla para continuar
*****

```

```

C:\Abc31\TMP\NUMEBUR2.EXE
*****
No de dato      Datos sin ordenar      Datos ordenados
*****
966             220                    6
968             80                     5
969             151                    5
970             139                    4
971             1                     4
972             80                     4
973             93                     4
974             186                    3
975             82                     3
976             1000                   3
977             18                     3
978             37                     3
979             24                     2
980             243                    2
981             164                    2
982             19                     2
983             85                     2
984             76                     1
985             64                     1
986             95                     1
987             125                    1
988             171                    1
989             164                    1
990             1000                   1
991             42                     1
992             69                     1
993             50                     1
994             119                    1
995             94                     0
996             254                    0
997             166                    0
998             10                     0
999             106                    0
1000            199                    0
*****

```

El tercer programa debe ordenar una cadena de caracteres por lo que sigue la misma metodología por que como sabemos en el código ASCII los caracteres son tratados como números, por lo tanto esta vez en vez de imprimir números mandaremos a imprimir caracteres lo cual es logradamente con el modificador de impresión %c en la impresión de las dos cadenas y dejamos normal al del número de casilla.

Por lo tanto nuestro programa quedo de la siguiente forma:

```

/*
FECHA: 07-NOV-06
PROGRAMA: 24
OBJETIVO: /* APLICAR EL METODO DE LA BURBUJA PARA LA ORDENACIÓN
DE CARACTERES EN VEZ DE HACERLO CON NUMEROS */
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include <conio.h>
#include <dos.h>
main ()
{
int i,j,u,x=1,l,g=5;
int a[1000],k,pasada,aux,b[1000];
textmode (C4350);
textcolor (RED);
textbackground (CYAN);
clrscr ();
for (l=1;l<=80;l++)
{
gotoxy (l,1);
printf ("*");
delay (g);
}
for (l=1;l<=49;l++)
{
gotoxy(80,l);

```

```

printf ("*");
delay (g);
}
for (l=80; l>=1;--l)
{
gotoxy (l,49);
printf ("*");
delay (g);
}
for (l=49; l>=1;--l)
{
gotoxy (1,l);
printf ("*");
delay (g);
}
randomize ();
for (j=0;j<1000;j++)
{
i=random (127);
//printf ("random number in the 0-255 range: %c %d \n\n",i,i);
//delay (2);
if (i>32)
{
a[j]=i;
b[j]=i;
}
else
{
j--;
}
}
for (pasada=1;pasada<=999;pasada++)
{
for (k=0;k<=998;k++)
{
if (a[k]>a[k+1])
{
aux=a[k];
a[k]=a[k+1];
a[k+1]=aux;
}
}
}
printf ("\n*\t No de dato\tDatos sin ordenar \t Datos ordenados\n\n");
for (k=0;k<1000;k++)
{
u=x*42;
if (k==u)
{
x++;
gotoxy (5,47);
printf ("Presiona una tecla para continuar");
getch ();
clrscr ();
for (l=1;l<=80;l++)
{

```

```

        gotoxy (1,1);
        printf ("*");
        delay (g);
    }
    for (l=1;l<=49;l++)
    {
        gotoxy(80,l);
        printf ("*");
        delay (g);
    }
    for (l=80; l>=1;--l)
    {
        gotoxy (l,49);
        printf ("*");
        delay (g);
    }
    for (l=49;l>=1;--l)
    {
        gotoxy (1,l);
        printf ("*");
        delay (g);
    }
    printf ("\n*\t No de dato\tDatos sin ordenar \t Datos ordenados");
    printf ("\n\n*\t\t%d\t\t%c\t\t%c\n",k,b[k],a[k]);
}
else
{
    printf ("*\t\t%d\t\t%c\t\t%c\n",k+1,b[k],a[k]);
}
}
getch ();
}
/*CONCLUSIONES: COMO SE NOTA EN EL CODIGO FUENTE EL METODO DE
ORDENACIÓN USADO ES EXACTAMENTE EL MISMO QUE EN EL PRIMER
PROGRAMA SOLO CAMBIA LA FORMA DE IMPRESIÓN Y EL RANGO DE
LLENADO DE LA CADENA YA QUE LOS CARACTERES SE PUEDEN VER
COMO NUMEROS EN EL CODIGO ASCII*/

```

La corrida se ve de la siguiente manera:

```

C:\Abc31\TMP\CARABUR.EXE
*****
No de dato      Datos sin ordenar      Datos ordenados
1              U                      t
2              B                      t
3              U                      t
4              5                      t
5              I                      t
6              a                      t
7              -                      t
8              C                      t
9              u                      t
10             v                      t
11             >                      t
12             x                      t
13             v                      t
14             B                      t
15             y                      "
16             i                      "
17             ?                      "
18             x                      "
19             u                      "
20             s                      "
21             P                      "
22             a                      "
23             i                      "
24             t                      "
25             h                      "
26             '                      "
27             U                      "
28             H                      "
29             4                      "
30             S                      "
31             T                      "
32             T                      "
33             c                      "
34             U                      "
35             P                      "
36             '                      "
37             -                      "
38             B                      "
39             -                      "
40             -                      "
41             -                      "
42             B                      "

Presiona una tecla para continuar...
*****

```

```

C:\Abc31\TMP\CARABUR.EXE
*****
No de dato      Datos sin ordenar      Datos ordenados
966             -                      <
968             6                      |
969             S                      |
970             B                      |
971             i                      |
972             -                      |
973             2                      |
974             x                      |
975             S                      |
976             W                      |
977             j                      |
978             N                      |
979             a                      >
980             -                      >
981             a                      >
982             x                      >
983             p                      >
984             -                      >
985             B                      >
986             a                      >
987             P                      >
988             h                      >
989             N                      >
990             +                      >
991             -                      ~
992             i                      ~
993             k                      ~
994             ^                      ~
995             s                      ~
996             t                      ~
997             j                      ~
998             P                      ~
999             ?                      ~
1000            J                      ~

*****

```

```

C:\Abc31\TMP\ICARABUR.EXE
*****
No de dato      Datos sin ordenar      Datos ordenados
*****
420             R                K
422             L                K
423             -                K
424             S                K
425             n                K
426             l                K
427             -                K
428             >                L
429             r                L
430             9                L
431             z                L
432             6                L
433             &                L
434             M                L
435             :                L
436             =                L
437             =                L
438             n                L
439             B                L
440             Q                M
441             n                M
442             -                M
443             B                M
444             x                M
445             :                M
446             y                M
447             T                M
448             C                M
449             B                M
450             G                M
451             -                M
452             p                N
453             l                N
454             P                N
455             s                N
456             o                N
457             -                N
458             c                N
459             2                N
460             E                N
461             6                O
462             L                O
*****
Presiona una tecla para continuar
*****

```

El ultimo programa debe realizar la ordenación de una cadena de caracteres en forma descendente por lo que usamos las características para imprimir caracteres del tercer programa y las especificaciones para ordenar de manera descendente del segundo programa, cabe mencionar que en este y en el anterior programa el random usa como limite el 127 y se excluyen los números anteriores al 33 ya que estos generan problemas con la impresión además de que algunos no muestran un carácter. Sabiendo todo lo anterior el código fuente del programa quedo de la siguiente forma:

```

/*
FECHA: 07-NOV-06
PROGRAMA: 25
OBJETIVO: /*APLICAR LOS CONOCIMIENTOS ADQUIRIDOS EN EL USO
DEL METODO DE LA BURBUJA PARA ORDENAR UNA CADENA DE CARACTERES
DE FORMA DESCENDENTE */
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include <conio.h>
#include <dos.h>
main ()
{
int i,j,u,x=1,l,g=5;
int a[1000],k,pasada,aux,b[1000];
textmode (C4350);
textcolor (RED);
textbackground (CYAN);
clrscr ();
for (l=1;l<=80;l++)
{
gotoxy (l,1);
printf ("%*");
delay (g);
}
for (l=1;l<=49;l++)

```

```

{
gotoxy(80,1);
printf ("*");
delay (g);
}
for (l=80;l>=1;--l)
{
gotoxy (l,49);
printf ("*");
delay (g);
}
for (l=49;l>=1;--l)
{
gotoxy (1,l);
printf ("*");
delay (g);
}
randomize ();
for (j=0;j<1000;j++)
{
i=random (127);
//printf ("random number in the 0-255 range: %c %d \n\n",i,i);
//delay (2);
if (i>32)
{
a[j]=i;
b[j]=i;
}
else
{
j--;
}
}
for (pasada=1;pasada<=999;pasada++)
{
for (k=0;k<=998;k++)
{
if (a[k]<a[k+1])
{
aux=a[k];
a[k]=a[k+1];
a[k+1]=aux;
}
}
}
printf ("\n*\t No de dato\tDatos sin ordenar \t Datos ordenados\n\n");
for (k=0;k<1000;k++)
{
u=x*42;
if (k==u)
{
x++;
gotoxy (5,47);
printf ("Presiona una tecla para continuar");
getch ();
clrscr ();
}
}

```

```

for (l=1;l<=80;l++)
{
gotoxy (l,1);
printf ("*");
delay (g);
}
for (l=1;l<=49;l++)
{
gotoxy(80,l);
printf ("*");
delay (g);
}
for (l=80; l>=1;--l)
{
gotoxy (l,49);
printf ("*");
delay (g);
}
for (l=49;l>=1;--l)
{
gotoxy (1,l);
printf ("*");
delay (g);
}
printf ("\n*\t No de dato\tDatos sin ordenar \t Datos ordenados");
printf ("\n\n*\t\t%d\t\t%c\t\t%c\n",k,b[k],a[k]);
}
else
{
printf ("*\t\t%d\t\t%c\t\t%c\n",k+1,b[k],a[k]);
}
}
getch ();
}
/*COMO SE NOTA EN EL CODIGO FUENTE SE VUELVE A REPETIR
EL METODO DE LA BURBUJA CON MODIFICACIONES HECHAS
EN EL TERCER PROGRAMA PARA QUE LA CADENA SEA DE
CARACTERES Y SE IMPRIMAN COMO TAL (%C) Y LA MODIFICACION
QUE SE LE HACE EN EL SEGUNDO PROGRAMA PARA QUE
ORDENE LA CADENA DE MANERA DESCENDENTE CON LO CUAL SE OBTIENE
ESTE NUEVO PROGRAMA*/

```

La corrida se ve de la siguiente forma:

```

C:\Abc31\TMP\CARABUR2.EXE
*****
No de dato      Datos sin ordenar      Datos ordenados
1              P                      ~
2              \                      ~
3              o                      ~
4              H                      ~
5              ?                      ~
6              8                      ~
7              x                      ~
8              z                      ~
9              -                      ~
10             v                      ~
11             i                      >
12             -                      >
13             M                      >
14             z                      >
15             >                      >
16             >                      >
17             G                      >
18             0                      >
19             z                      >
20             h                      >
21             e                      >
22             n                      >
23             o                      >
24             G                      >
25             t                      |
26             y                      |
27             I                      |
28             -                      |
29             G                      |
30             k                      |
31             /                      |
32             h                      |
33             *                      |
34             r                      |
35             S                      <
36             ?                      <
37             *                      <
38             <                      <
39             I                      <
40             G                      <
41             9                      <
42             Q                      <

Presiona una tecla para continuar
*****

```

```

C:\Abc31\TMP\CARABUR2.EXE
*****
No de dato      Datos sin ordenar      Datos ordenados
420             H                      U
422             a                      U
423             i                      U
424             -                      U
425             P                      U
426             E                      U
427             m                      U
428             <                      U
429             S                      U
430             R                      U
431             i                      U
432             d                      U
433             w                      U
434             f                      U
435             l                      U
436             *                      U
437             "                      U
438             t                      U
439             C                      I
440             R                      I
441             E                      I
442             N                      I
443             ?                      I
444             A                      I
445             >                      I
446             M                      T
447             B                      S
448             G                      S
449             U                      S
450             v                      S
451             -                      S
452             z                      S
453             M                      S
454             n                      S
455             C                      S
456             t                      S
457             l                      S
458             )                      S
459             ?                      S
460             ?                      S
461             u                      S
462             N                      R

Presiona una tecla para continuar
*****

```

```

C:\bc31\TMP\CARABUR2.EXE
*****
No de dato      Datos sin ordenar      Datos ordenados
*****
966             :                       5
968             S                       8
969             c                       3
970             '                       9
971             n                       6
972             E                       7
973             m                       4
974             >                       2
975             ~                       1
976             j                       0
977             )                       11
978             z                       12
979             >                       11
980             2                       11
981             b                       11
982             L                       11
983             u                       11
984             S                       11
985             (                       11
986             '                       11
987             l                       1
988             >                       1
989             S                       1
990             7                       1
991             =                       1
992             2                       1
993             ~                       1
994             q                       1
995             a                       1
996             0                       1
997             B                       1
998             H                       1
999             '                       1
1000            _                       1
*****

```

**Conclusiones generales:**

En esta practica aprendimos a usar el método de la burbuja el cual es bastante funcional para ordenar cadenas ya sea de números y de caracteres y sigue prácticamente la misma plantilla para ordenar tanto descendente como ascendentemente además de que nuevamente vemos la importancia de los ciclos de repetición for ya que gracias a ellos es que se puede ordenar la cadena