

ESCUELA POLITÉCNICA
NACIONAL
PROGRAMACIÓN II
COMPONENTE PRÁCTICO

DOCENTE: Eddie Hans Yáñez Quezada

ESTUDIANTE: Josune Antonella Singaña Tapia

FECHA: 31 de Julio de 2022

ARRREGLOS
DEBER 7



Escuela Politécnica Nacional
Facultad De Ingeniería en Sistemas
PROGRAMACIÓN I (COMPONENTE PRÁCTICO)



DOCENTE: Eddie Hans Yáñez Quezada

ESTUDIANTE: Josune Antonella Singaña Tapia

TEMA: Arreglos

FECHA: 31 de julio de 2022

INDICE

Objetivos

Problema 1

Problema 2

Problema 3

Problema 4

Problema 5

Problema 6

Problema 7

Problema 8

Problema 9

OBJETIVOS DEL DEBER:

- Comprender el uso de arreglos para la resolución de ejercicios.
- Analizar la estructura en lenguaje C del algoritmo desarrollado en pseint.
- Resolver los ejercicios impares de la Página 267 del libro de Chaves

ACTIVIDAD EJERCICIO DE ALGORITMOS

Problema 1

1

Desarrollo			Etapa
<p>Dados dos vectores numéricos diseñar un algoritmo que identifique y muestre los números que tienen en común. (Nota: la dimensión de los vectores puede ser igual o distinta)</p>			Definición del problema
Entrada	Proceso	Salida	Análisis del problema
<p>Vec1, vec2</p> <p>E enteros.</p> <p>Resp E carácter</p> <p>vector1[x], vector2[x] E enteros</p>	<p>Números al azar</p> <p>Para x=1 hasta vec1 Hacer</p> <p style="padding-left: 40px;">vector1[x]=azar(10)+1</p> <p style="padding-left: 40px;">j=vector1[x]</p> <p>FinPara</p> <p>Para t=1 hasta vec2 Hacer</p> <p style="padding-left: 40px;">vector2[t]=azar(10)+1</p> <p>FinPara</p> <p>Para x=1 hasta vec1 Hacer</p> <p style="padding-left: 40px;">j=vector1[x]</p> <p style="padding-left: 40px;">Para i=1 hasta vec2 Hacer</p> <p style="padding-left: 80px;">Si j = vector2[i] Entonces</p> <p style="padding-left: 120px;">num[x]= j</p> <p style="padding-left: 80px;">FinSi</p> <p style="padding-left: 40px;">FinPara</p> <p>FinPara</p> <p>Ingreso de elementos por teclado</p> <p>Para x=0 hasta vec1-1 Hacer</p> <p style="padding-left: 40px;">Escribir "Ingrese el elemento [, x+1 "]</p> <p>del primer vector" Sin Saltar</p> <p style="padding-left: 40px;">Leer vector1[x]</p> <p style="padding-left: 40px;">j=vector1[x]</p> <p>FinPara</p> <p>Para i=0 hasta vec2-1 Hacer</p> <p style="padding-left: 40px;">Escribir "Ingrese el elemento [, i+1 "]</p> <p>del segundo vector" Sin Saltar</p> <p style="padding-left: 40px;">Leer vector2[i]</p> <p>FinPara</p> <p>Para x=0 hasta vec1-1 Hacer</p> <p style="padding-left: 40px;">j=vector1[x]</p> <p style="padding-left: 40px;">Para i=0 hasta vec2-1 Hacer</p> <p style="padding-left: 80px;">Si j = vector2[i] Entonces</p> <p style="padding-left: 120px;">num[x]= j</p> <p style="padding-left: 80px;">FinSi</p> <p style="padding-left: 40px;">FinPara</p> <p>FinPara</p>	<p>Num E enteros</p>	
<p>Imagen obtenida en pseint</p> <pre>// autor: Josune Singaña // fecha: 31-07-2022 Algoritmo dosvectores Definir vec1 Como Entero Definir vec2 Como Entero Definir resp Como Caracter Escribir " APLICACION COMPARA LOS ELEMENTOS DE DOS VECTORES " Escribir "INGRESE EL NUMERO DE ELEMENTOS DEL PRIMER VECTOR: " Sin Saltar Leer vec1 Escribir "INGRESE EL NUMERO DE ELEMENTOS DEL SEGUNDO VECTOR: " Sin Saltar Leer vec2 Escribir "DESEA QUE LOS ELEMENTOS SE GENEREN ALEATORIAMENTE DEL 1 AL 10: s o n " Sin Saltar Leer resp Si resp = 's' o resp ='S' Entonces si vec1 ≤ vec2 Entonces arreglo1(vec1,vec2)</pre>			Diseño del algoritmo

```

SiNo
..... arreglo1(vec2,vec1)
FinSi
SiNo
..... si vec1 ≤ vec2 Entonces
..... arreglo2(vec1,vec2)
..... SiNo
..... arreglo2(vec2,vec1)
..... FinSi
FinSi
FinAlgoritmo
Funcion arreglo2(vec1,vec2)
..... Definir vector1 Como Entero
..... Definir vector2 Como Entero
..... Definir num Como Entero
..... Definir resp Como Caracter
..... Dimension vector1[100]
..... Dimension vector2[100]
..... Dimension num[100]
..... Definir i Como Entero
..... Definir j Como Entero
..... Definir x Como Entero
..... Para x=1 hasta vec1 Hacer
..... .. Escribir "Ingrese el elemento [", x "]" del primer vector" Sin Saltar
..... .. Leer vector1[x]
..... .. j=vector1[x]
..... FinPara
..... Para i=1 hasta vec2 Hacer
..... .. Escribir "Ingrese el elemento [", i "]" del segundo vector" Sin Saltar
..... .. Leer vector2[i]
..... FinPara
..... Para x=1 hasta vec1 Hacer
..... .. j=vector1[x]
..... .. Para i=1 hasta vec2 Hacer
..... .. .. Si j = vector2[i] Entonces
..... .. .. .. num[x]= j
..... .. .. FinSi
..... .. FinPara
..... FinPara
..... resultados(vec1,vector1,vec2,vector2,num)
FinFuncion
Funcion resultados(vec1,vector1,vec2,vector2,num)
..... Definir i Como Entero
..... Escribir "El PRIMER VECTOR ES: { " Sin Saltar
..... Para i=1 hasta vec1 Hacer
..... .. Escribir vector1[i] " " Sin Saltar
..... FinPara
..... Escribir "}"
..... Escribir "El SEGUNDO VECTOR ES: { " Sin Saltar
..... Para i=1 hasta vec2 Hacer
..... .. Escribir vector2[i] " " Sin Saltar
..... FinPara
..... Escribir "}"
..... Escribir "LOS VALORES QUE SE REPITEN SON: { " Sin Saltar
..... Para i=1 hasta vec1 Hacer
..... .. Si num[i] ≠ 0 Entonces
..... .. .. Escribir num[i] " " Sin Saltar
..... .. FinSi
..... .. Escribir " " Sin Saltar
..... FinPara
..... Escribir "}"
FinFuncion

```

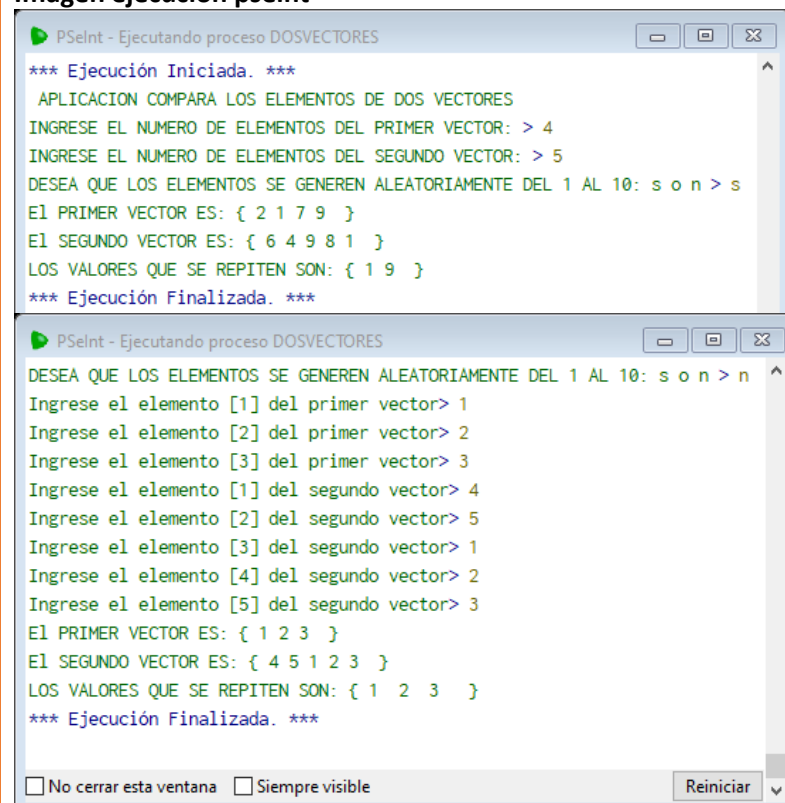


```

Funcion arreglar [vec1,vec2]
  Definir vector1 Como Entero
  Definir vector2 Como Entero
  Definir num Como Entero
  Definir resp Como Caracter
  Dimension vector1[100]
  Dimension vector2[100]
  Dimension num[100]
  Definir i Como Entero
  Definir j Como Entero
  Definir x Como Entero
  Definir t Como Entero
  Para x=1 hasta vec1 Hacer
    vector1[x]=azar(10)+1
    j=vector1[x]
  FinPara
  Para t=1 hasta vec2 Hacer
    vector2[t]=azar(10)+1
  FinPara
  Para x=1 hasta vec1 Hacer
    j=vector1[x]
    Para i=1 hasta vec2 Hacer
      Si j = vector2[i] Entonces
        num[x]= j
      FinSi
    FinPara
  FinPara
  resultados(vec1,vector1,vec2,vector2,num)
FinFuncion

```

Imagen ejecución pseint



**Verificación
del algoritmo**

Imagen obtenida en Dev c++

```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
#include<conio2.h>

void arreglo2(int vec1, int vec2);
void resultados(int vec1, int vector1[], int vec2, int vector2[], int num[]);
void arreglo1(int vec1, int vec2);
/* autor: Josune Singaña */
/* fecha:31-07-2022 */
int main() {
    char resp;
    int vec1, vec2;
    textbackground(YELLOW);
    clrscr();
    textcolor(MAGENTA);
    gotoxy (25,2);printf(" APLICACION COMPARA LOS ELEMENTOS DE DOS VECTORES \n");
    textcolor(CYAN);
    gotoxy (15,4);printf("INGRESE EL NUMERO DE ELEMENTOS DEL PRIMER VECTOR: ");
    scanf("%i",&vec1);
    gotoxy (15,5);printf("INGRESE EL NUMERO DE ELEMENTOS DEL SEGUNDO VECTOR: ");
    scanf("%i",&vec2);
    textcolor(BLUE);
    gotoxy (15,7);printf("DESEA QUE LOS ELEMENTOS SE GENEREN ALEATORIAMENTE DEL 1 AL 10: s o n ");
    resp = getch();
    if (resp=='s' || resp=='S') {
        if (vec1<=vec2) {
            arreglo1(vec1,vec2);
        } else {
            arreglo1(vec2,vec1);
        }
    } else {
        if (vec1<=vec2) {
            arreglo2(vec1,vec2);
        } else {
            arreglo2(vec2,vec1);
        }
    }
    return 0;
}

void arreglo2(int vec1, int vec2) {
    textcolor(6);
    int i, j, x, n, num[100], vector1[100], vector2[100];
    for (n=1;n<100;n++) {
        num[n]=0;
    }
    for (x=1;x<=vec1;x++) {
        printf("\nIngrese el elemento [%i] del primer vector: ",x);
        scanf("%i",&vector1[x]);
        j = vector1[x];
    }
    for (i=1;i<=vec2;i++) {
        printf("Ingrese el elemento [%i] del segundo vector: ",i);
        scanf("%i",&vector2[i]);
    }
    for (x=1;x<=vec1;x++) {
        j = vector1[x];
        for (i=1;i<=vec2;i++) {
            if (j==vector2[i]) {
                num[x] = j;
            }
        }
    }
    resultados(vec1,vector1,vec2,vector2,num);
}

void resultados(int vec1, int vector1[], int vec2, int vector2[], int num[]) {
    int i;
    textcolor(8);
    printf("\n EL PRIMER VECTOR ES: { ");
```

Diseño del
algoritmo

```

        for (i=1;i<=vec1;i++) {
            printf("%i ",vector1[i]);
        }
        printf(" }\n");
        printf("El SEGUNDO VECTOR ES: { ");
        for (i=1;i<=vec2;i++) {
            printf("%i ",vector2[i]);
        }
        printf(" }\n");
        printf("LOS VALORES QUE SE REPITEN SON: { ");
        for (i=1;i<=vec1;i++) {
            if (num[i]!=0) {
                printf("%i ",num[i]);
            }
        }
        printf(" ");
        printf(" }\n");
    }

void arreglo1(int vec1, int vec2) {
    int i, j, x, t, n, num[100];
    int vector1[100], vector2[100];
    for (n=1;n<100;n++) {
        num[n]=0;
    }
    for (x=1;x<=vec1;x++) {
        vector1[x] = (rand()%10)+1;
        j = vector1[x];
    }
    for (t=1;t<=vec2;t++) {
        vector2[t] = (rand()%10)+1;
    }
    for (x=1;x<=vec1;x++) {
        j = vector1[x];
        for (i=1;i<=vec2;i++) {
            if (j==vector2[i]) {
                num[x] = j;
            }
        }
    }
    resultados(vec1,vector1,vec2,vector2,num);
}

```

Imagen ejecución Dev C++

C:\Users\Home\Documents\1SEMESTRE\PERIODO 2022 A\PROGRAMACION 1\ALGORITMOS\DEBERES\Deber 7\ejercicio C\d

APLICACION COMPARA LOS ELEMENTOS DE DOS VECTORES

INGRESE EL NUMERO DE ELEMENTOS DEL PRIMER VECTOR: 3
INGRESE EL NUMERO DE ELEMENTOS DEL SEGUNDO VECTOR: 4

DESEA QUE LOS ELEMENTOS SE GENEREN ALEATORIAMENTE DEL 1 AL 10: s o n

Ingrese el elemento [1] del primer vector: 1
Ingrese el elemento [2] del primer vector: 2
Ingrese el elemento [3] del primer vector: 3
Ingrese el elemento [1] del segundo vector: 32
Ingrese el elemento [2] del segundo vector: 2
Ingrese el elemento [3] del segundo vector: 3
Ingrese el elemento [4] del segundo vector: 9

El PRIMER VECTOR ES: { 1 2 3 }
El SEGUNDO VECTOR ES: { 32 2 3 9 }
LOS VALORES QUE SE REPITEN SON: { 2 3 }

C:\Users\Home\Documents\1SEMESTRE\PERIODO 2022 A\PROGRAMACION 1\ALGORITMOS\DEBERES\Deber 7\ejercicio C\dosvect

APLICACION COMPARA LOS ELEMENTOS DE DOS VECTORES

INGRESE EL NUMERO DE ELEMENTOS DEL PRIMER VECTOR: 4
INGRESE EL NUMERO DE ELEMENTOS DEL SEGUNDO VECTOR: 3

DESEA QUE LOS ELEMENTOS SE GENEREN ALEATORIAMENTE DEL 1 AL 10: s o n

El PRIMER VECTOR ES: { 2 8 5 }
El SEGUNDO VECTOR ES: { 1 10 5 9 }
LOS VALORES QUE SE REPITEN SON: { 5 }

Process exited after 10.38 seconds with return value 0
Presione una tecla para continuar . . .

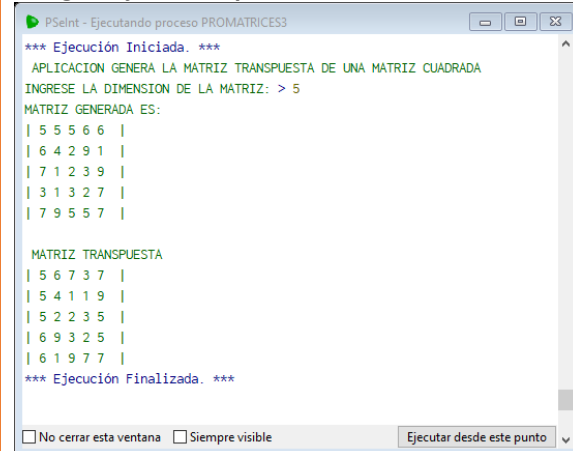
**Verificación
del
algoritmo**

Problema 2

3

Desarrollo			Etapa
Diseñar un algoritmo para trasponer una matriz cuadrada			Definición del problema
Entrada dim E enteros	Proceso Generar matriz para i=1 hasta dim Hacer para j=1 Hasta dim Hacer mat[i,j]=azar[9]+1 FinPara FinPara Transpuesta: Para j=1 hasta dim Hacer Escribir " " Sin Saltar para i=1 Hasta dim Hacer Escribir mat[i,j]," "Sin Saltar FinPara Escribir " " FinPara	Salida mat[i,j] E entero	Análisis del problema
Imagen obtenida en pseint <pre>// autor: Josune Singaña //fecha:31-07-2022 Algoritmo promatrices3 definir i Como Entero Definir j Como Entero Definir mat Como Entero Definir dim Como Entero Dimension mat[100,100] Escribir " APLICACION GENERA LA MATRIZ TRANSPUESTA DE UNA MATRIZ CUADRADA" Escribir "INGRESE LA DIMENSION DE LA MATRIZ: " Sin Saltar Leer dim para i=1 hasta dim Hacer para j=1 Hasta dim Hacer mat[i,j]=azar[9]+1 FinPara FinPara Escribir "MATRIZ GENERADA ES: " Para i=1 Hasta dim Hacer Escribir " " Sin Saltar para j=1 hasta dim Hacer Escribir mat[i,j]," "Sin Saltar FinPara Escribir " " FinPara Escribir "" Escribir " MATRIZ TRANSPUESTA" Para j=1 hasta dim Hacer Escribir " " Sin Saltar para i=1 Hasta dim Hacer Escribir mat[i,j]," "Sin Saltar FinPara Escribir " " FinPara FinAlgoritmo</pre>			Diseño del algoritmo

Imagen ejecución pseint



```
*** Ejecución Iniciada. ***
APLICACION GENERA LA MATRIZ TRANSPUESTA DE UNA MATRIZ CUADRADA
INGRESE LA DIMENSION DE LA MATRIZ: > 5
MATRIZ GENERADA ES:
| 5 5 5 6 6 |
| 6 4 2 9 1 |
| 7 1 2 3 9 |
| 3 1 3 2 7 |
| 7 9 5 5 7 |

MATRIZ TRANSPUESTA
| 5 6 7 3 7 |
| 5 4 1 1 9 |
| 5 2 2 3 5 |
| 6 9 3 2 5 |
| 6 1 9 7 7 |
*** Ejecución Finalizada. ***
```

Verificación del
algoritmo

Imagen obtenida en Dev c++

```
#include<stdio.h>
#include<stdlib.h>
#include<conio2.h>
/* autor: Josune Singaña */
/* fecha:31-07-2022 */
int main() {
    int dim, i, j, mat[100][100];
    textbackground(0);
    clrscr();
    textcolor(14);
    gotoxy (25,2);printf(" APLICACION GENERA LA MATRIZ TRANSPUESTA DE UNA MATRIZ CUADRADA\n");
    textcolor(10);
    gotoxy (15,5);printf("INGRESE LA DIMENSION DE LA MATRIZ: ");
    scanf("%i",&dim);
    for (i=1;i<=dim;i+=1) {
        for (j=1;j<=dim;j+=1) {
            mat[i][j] = (rand()%9)+1;
        }
        textcolor(9);
        printf("MATRIZ GENERADA ES: \n");
        for (i=1;i<=dim;i+=1) {
            printf("| ");
            for (j=1;j<=dim;j+=1) {
                printf("%i ",mat[i][j]);
            }
            printf(" |\n");
        }
        printf("\n");
        textcolor(11);
        printf(" MATRIZ TRANSPUESTA\n");
        for (j=1;j<=dim;j+=1) {
            printf("| ");
            for (i=1;i<=dim;i+=1) {
                printf("%i ",mat[i][j]);
            }
            printf(" |\n");
        }
        return 0;
    }
}
```

Diseño del
algoritmo

Imagen ejecución Dev C++

The image displays three sequential screenshots of a C++ application titled "APLICACION GENERA LA MATRIZ TRANSPUESTA DE UNA MATRIZ CUADRADA". Each screenshot shows the program's execution for a different matrix dimension, with the original matrix and its transpose displayed side-by-side.

First Screenshot (Dimension 7):

```
INGRESE LA DIMENSION DE LA MATRIZ: 7
MATRIZ GENERADA ES:
6 9 8 5 9 2 4 |
1 8 3 9 3 8 7 |
8 6 8 9 4 1 1 |
7 6 1 5 8 7 6 |
9 6 3 1 3 1 7 |
5 9 2 8 4 3 7 |
3 4 7 3 4 8 3 |

MATRIZ TRANSPUESTA
6 1 8 7 9 5 3 |
9 8 6 6 6 9 4 |
8 3 8 1 3 2 7 |
5 9 9 5 1 8 3 |
9 3 4 8 3 4 4 |
2 8 1 7 1 3 8 |
4 7 1 6 7 7 3 |

Process exited after 2.427 seconds with return value 0
Presione una tecla para continuar . . .
```

Second Screenshot (Dimension 9):

```
INGRESE LA DIMENSION DE LA MATRIZ: 9
MATRIZ GENERADA ES:
6 9 8 5 9 2 4 1 8 |
3 9 3 8 7 8 6 8 9 |
4 1 1 7 6 1 5 8 7 |
6 9 6 3 1 3 1 7 5 |
9 2 8 4 3 7 3 4 7 |
3 4 8 3 2 6 6 2 7 |
4 8 3 4 8 5 5 3 6 |
7 1 2 5 6 5 5 6 1 |
6 7 8 6 4 7 4 3 1 |

MATRIZ TRANSPUESTA
6 3 4 6 9 3 4 7 6 |
9 9 1 9 2 4 8 1 7 |
8 3 1 6 8 8 3 2 8 |
5 8 7 3 4 3 4 5 6 |
9 7 6 1 3 2 8 6 4 |
2 8 1 3 7 6 5 5 7 |
4 6 5 1 3 6 5 5 4 |
1 8 8 7 4 2 3 6 3 |
8 9 7 5 7 7 6 1 1 |

Process exited after 3.699 seconds with return value 0
Presione una tecla para continuar . . .
```

Third Screenshot (Dimension 2):

```
INGRESE LA DIMENSION DE LA MATRIZ: 2
MATRIZ GENERADA ES:
6 9 |
8 5 |

MATRIZ TRANSPUESTA
6 8 |
9 5 |

Process exited after 4.799 seconds with return value 0
Presione una tecla para continuar . . .
```

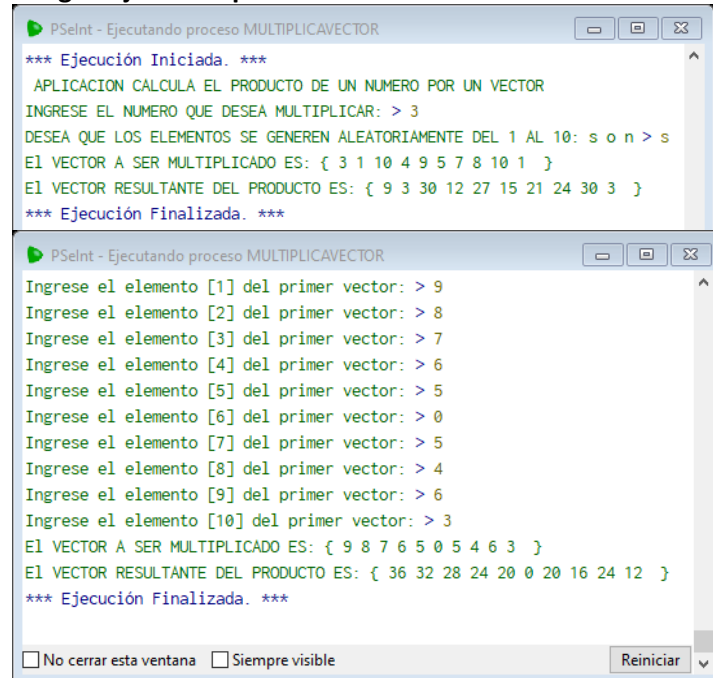
Verificación
del
algoritmo

Problema 3

5

Desarrollo			Etapa
Diseñar un algoritmo para calcular el producto de un número por un vector numérico de 10 elementos			Definición del problema
Entrada	Proceso	Salida	Análisis del problema
num E enteros. vector[i] E enteros	<p>Con elementos al azar Para i=0 hasta 9 Hacer vector[i]= azar(10)+1 j=vector[i] resul[i] = num * j FinPara</p> <p>Con ingreso de elementos por teclado Para i=1 hasta 10 Hacer Escribir "Ingrese el elemento [, i]" del primer vector: " Sin Saltar Leer vector[i] j=vector[i] resul[i] = num * j FinPara</p>	resul[i] E enteros	
<p>Imagen obtenida en pseint</p> <pre> // autor: Josune Singaña // fecha: 31-07-2022 Algoritmo multiplicavector Definir resp Como Caracter definir i Como Entero Definir j Como Entero Definir resul Como Entero Definir vector Como Entero Definir num Como Entero Dimension vector[10] Dimension resul[10] Escribir " APLICACION CALCULA EL PRODUCTO DE UN NUMERO POR UN VECTOR " Escribir "INGRESE EL NUMERO QUE DESEA MULTIPLICAR: " Sin Saltar Leer num Escribir "DESEA QUE LOS ELEMENTOS SE GENEREN ALEATORIAMENTE DEL 1 AL 10: s o n " Sin Saltar Leer resp Si resp = 's' o resp = 'S' Entonces Para i=0 hasta 9 Hacer vector[i]=azar(10)+1 j=vector[i] resul[i] = num * j FinPara Sino Para i=0 hasta 9 Hacer Escribir "Ingrese el elemento [, i+1]" del primer vector: " Sin Saltar Leer vector[i] j=vector[i] resul[i] = num * j FinPara Fin Si Escribir "El VECTOR A SER MULTIPLICADO ES: { " Sin Saltar Para i=0 hasta 9 Hacer Escribir vector[i] " " Sin Saltar FinPara Escribir " }" Escribir "El VECTOR RESULTANTE DEL PRODUCTO ES: { " Sin Saltar Para i=0 hasta 9 Hacer Escribir resul[i] " " Sin Saltar FinPara Escribir " }" FinAlgoritmo </pre>			Diseño del algoritmo

Imagen ejecución pseint



```
*** Ejecución Iniciada. ***
  APLICACION CALCULA EL PRODUCTO DE UN NUMERO POR UN VECTOR
  INGRESE EL NUMERO QUE DESEA MULTIPLICAR: > 3
  DESEA QUE LOS ELEMENTOS SE GENEREN ALEATORIAMENTE DEL 1 AL 10: s o n > s
  EL VECTOR A SER MULTIPLICADO ES: { 3 1 10 4 9 5 7 8 10 1 }
  EL VECTOR RESULTANTE DEL PRODUCTO ES: { 9 3 30 12 27 15 21 24 30 3 }
*** Ejecución Finalizada. ***

PSeInt - Ejecutando proceso MULTIPLICAVECTOR

Ingrese el elemento [1] del primer vector: > 9
Ingrese el elemento [2] del primer vector: > 8
Ingrese el elemento [3] del primer vector: > 7
Ingrese el elemento [4] del primer vector: > 6
Ingrese el elemento [5] del primer vector: > 5
Ingrese el elemento [6] del primer vector: > 0
Ingrese el elemento [7] del primer vector: > 5
Ingrese el elemento [8] del primer vector: > 4
Ingrese el elemento [9] del primer vector: > 6
Ingrese el elemento [10] del primer vector: > 3
EL VECTOR A SER MULTIPLICADO ES: { 9 8 7 6 5 0 5 4 6 3 }
EL VECTOR RESULTANTE DEL PRODUCTO ES: { 36 32 28 24 20 0 20 16 24 12 }
*** Ejecución Finalizada. ***

☐ No cerrar esta ventana ☐ Siempre visible Reiniciar
```

Verificación del algoritmo

Imagen obtenida en Dev c++

```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
#include<conio2.h>
/* autor: Jasune Singaña */
/* fecha:31-07-2022 */
int main() {
    int i, j, num, resul[10], vector[10];
    char resp;
    textbackground(4);
    clrscr();
    textcolor(14);
    gotoxy (25,2);printf(" APLICACION CALCULA EL PRODUCTO DE UN NUMERO POR UN VECTOR \n");
    textcolor(7);
    gotoxy (15,4);printf("INGRESE EL NUMERO QUE DESEA MULTIPLICAR: ");
    scanf("%i",&num);
    textcolor(11);
    gotoxy (15,6);printf("DESEA QUE LOS ELEMENTOS SE GENEREN ALEATORIAMENTE DEL 1 AL 10: s o n ");
    resp = getch();
    if (resp=='s' || resp=='S') {
        for (i=0;i<=9;i++) {
            vector[i] = (rand()%10)+1;
            j = vector[i];
            resul[i] = num*j;
        }
    }
    else {
        textcolor(4);
        for (i=0;i<=9;i++) {
            printf("Ingrese el elemento [%i] del primer vector: ",i+1);
            scanf("%i",&vector[i]);
            j = vector[i];
            resul[i] = num*j;
        }
    }
    textcolor(15);
    printf("\n\n      EL VECTOR A SER MULTIPLICADO ES: { ");
    for (i=0;i<=9;i++) {
        printf("%i ",vector[i]);
    }
    printf(" }\n");
    printf("\n      EL VECTOR RESULTANTE DEL PRODUCTO ES: { ");
    for (i=0;i<=9;i++) {
        printf("%i ",resul[i]);
    }
    printf(" }\n");
    getch();
    return 0;
}
```

Diseño del algoritmo

Imagen ejecución Dev C++

```
APLICACION CALCULA EL PRODUCTO DE UN NUMERO POR UN VECTOR
INGRESE EL NUMERO QUE DESEA MULTIPLICAR: 3
DESEA QUE LOS ELEMENTOS SE GENEREN ALEATORIAMENTE DEL 1 AL 10: s o n
El VECTOR A SER MULTIPLICADO ES: { 2 8 5 1 10 5 9 3 5 }
El VECTOR RESULTANTE DEL PRODUCTO ES: { 6 24 15 3 30 15 27 9 15 }
-----
Process exited after 10.28 seconds with return value 0
Presione una tecla para continuar . . .

APLICACION CALCULA EL PRODUCTO DE UN NUMERO POR UN VECTOR
INGRESE EL NUMERO QUE DESEA MULTIPLICAR: 4
DESEA QUE LOS ELEMENTOS SE GENEREN ALEATORIAMENTE DEL 1 AL 10: s o n
Ingrese el elemento [1] del primer vector: 4
Ingrese el elemento [2] del primer vector: 2
Ingrese el elemento [3] del primer vector: 1
Ingrese el elemento [4] del primer vector: 4
Ingrese el elemento [5] del primer vector: 2
Ingrese el elemento [6] del primer vector: 4
Ingrese el elemento [7] del primer vector: 1
Ingrese el elemento [8] del primer vector: 2
Ingrese el elemento [9] del primer vector: 4
Ingrese el elemento [10] del primer vector: 1
El VECTOR A SER MULTIPLICADO ES: { 4 2 1 4 2 4 1 2 4 1 }
El VECTOR RESULTANTE DEL PRODUCTO ES: { 16 8 4 16 8 16 4 8 16 4 }
-----
Process exited after 20.29 seconds with return value 0
Presione una tecla para continuar . . .
```

Verificación
del
algoritmo

Problema 4

7

Desarrollo			Etapas
Diseñar un algoritmo para insertar un dato en un vector en una posición escogida por el usuario, si la posición está ocupada los datos se desplazan a la derecha para dar espacio al nuevo dato. Si el vector está lleno no se inserta el dato y se muestra un mensaje.			Definición del problema
Entrada Dim, i, E enteros vector[i] E texto	Proceso Para j=1 hasta dim Hacer Escribir "INGRESE LA POSICION DEL DATO [", j "]: " Sin Saltar Leer i ultimo= vector[dim-1] Si i=x Entonces Para n=dim-2 Hasta 0 con paso -1 Hacer ultimo= vector[n] Para n=dim-1 Hasta 0 con paso -1 Hacer vector[n+1] = ultimo FinPara FinPara	Salida vector[i] E texto	Análisis del problema


```
PSeInt - Ejecutando proceso DATO
```

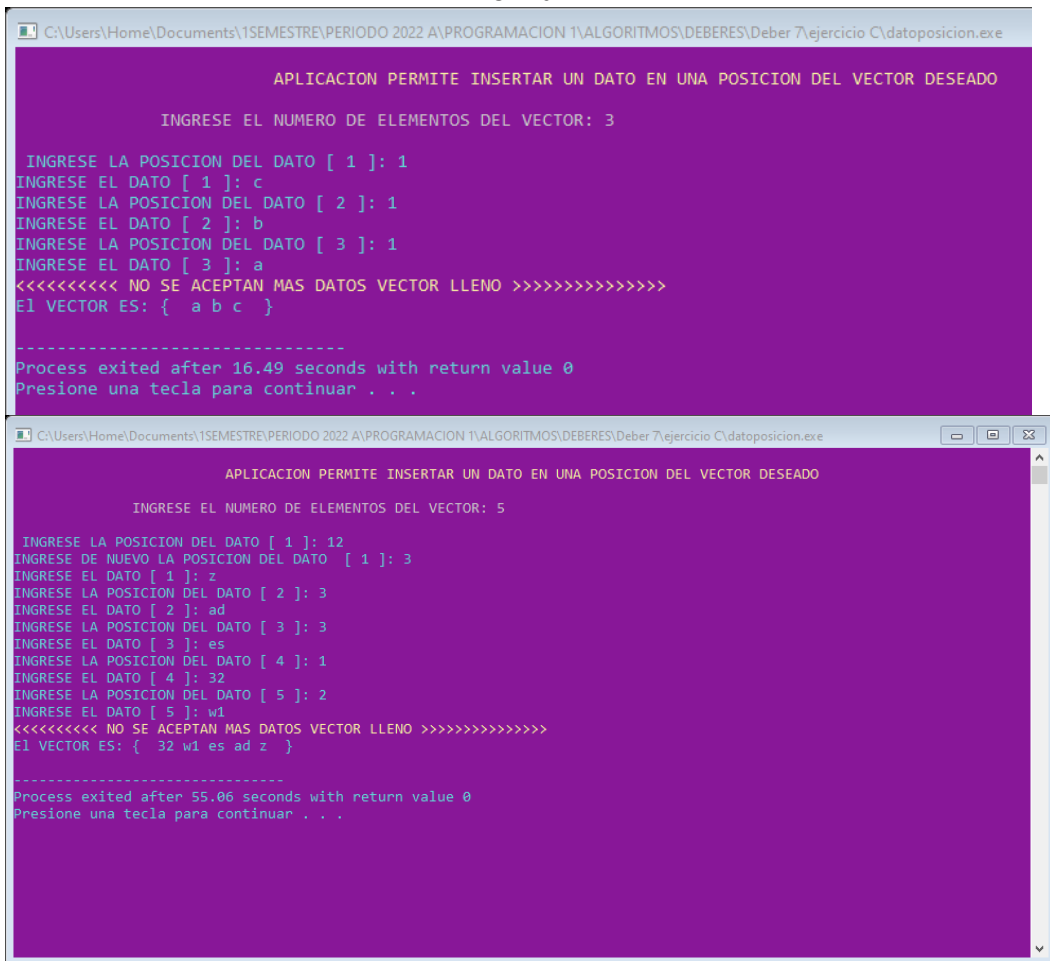
```
*** Ejecución Iniciada. ***  
    APLICACION PERMITE INSERTAR UN DATO EN UNA POSICION DEL VECTOR DESEADO  
INGRESE EL NUMERO DE ELEMENTOS DEL VECTOR: > 5  
INGRESE LA POSICION DEL DATO [ 1 ]: > 12  
INGRESE DE NUEVO LA POSICION DEL DATO [ 1 ]: > 3  
INGRESE EL DATO [ 1 ]: > z  
INGRESE LA POSICION DEL DATO [ 2 ]: > 3  
INGRESE EL DATO [ 2 ]: > es  
INGRESE LA POSICION DEL DATO [ 3 ]: > 3  
INGRESE EL DATO [ 3 ]: > ad  
INGRESE LA POSICION DEL DATO [ 4 ]: > 1  
INGRESE EL DATO [ 4 ]: > 32  
INGRESE LA POSICION DEL DATO [ 5 ]: > 2  
INGRESE EL DATO [ 5 ]: > w1  
<<<<<<<<<< NO SE ACEPTAN MAS DATOS VECTOR LLENO >>>>>>>>>>>>>>  
El Vector ES: { 32 w1 ad es z }  
*** Ejecución Finalizada. ***
```

Imagen obtenida en Dev c++

```
#include<stdio.h>
#include<ctype.h>
#include<string.h>
#include<conio2.h>
/* autor: Josune Singaña */
/* fecha:31-07-2022 */
int main() {
    int dim, i, j,x, n;
    char vector[20][40], ultimo[20];
    textbackground(MAGENTA);
    clrscr();
    textcolor(14);
    gotoxy(25,2); printf(" APLICACION PERMITE INSERTAR UN DATO EN UNA POSICION DEL VECTOR DESEADO\n");
    textcolor(7);
    gotoxy(15,4);printf("INGRESE EL NUMERO DE ELEMENTOS DEL VECTOR: ");
    scanf("%i",&dim);
    for (n=0;n<dim;n++) {
        strcpy( vector[n], "");
    }
    textcolor(11);
    printf("\n ");
    for (j=1;j<=dim;j++) {
        printf("INGRESE LA POSICION DEL DATO [ %i ]: ",j);
        scanf("%i",&i);
        while (i>dim) {
            printf("INGRESE DE NUEVO LA POSICION DEL DATO [ %i ]: ",j);
            scanf("%i",&i);
        }
        strcpy(ultimo, vector[dim-1]);
        if (i==x) {
            for (n=dim-2;n>=0;n--) {
                strcpy( ultimo , vector[dim]);
                for (n=dim-1;n>=0;n--) {
                    strcpy( vector[n+1],vector[n]);
                }
            }
            strcpy( vector[0], ultimo);
        }
        printf("INGRESE EL DATO [ %i ]: ",j);
        scanf("%s",vector[i]);
        x = i;
    }
    textcolor(14);
    printf("===== NO SE ACEPTAN MAS DATOS VECTOR LLENO =====\n");
    textcolor(11);
    printf("El VECTOR ES: ( ");
    for (i=0;i<=dim;i++) {
        printf("%s ",vector[i]);
    }
    printf(" )\n");
    getch();
    return 0;
}
```

Diseño del algoritmo

Imagen ejecución Dev C++



Verificación del algoritmo

Problema 5

9

Desarrollo			Etapa	
<p> Diseñar un algoritmo para hacer circular los datos de un vector: todos los elementos se desplazan una posición y el último elemento pasa a ocupar la primera posición. </p>			<p>Definición del problema</p>	
Entrada	Proceso	Salida	<p>Análisis del problema</p>	
<p> Numelem E enteros lista[n] texto </p>	<pre> Escribir "INGRESE EL NUMERO DE ELEMENTOS DE LA LISTA " Leer numelem Para n=0 hasta numelem-1 Hacer Escribir "INGRESA NOMBRE ", n+1 " " Leer lista[n] FinPara resp='s' Mientras Minusculas(resp)='s' Hacer temp=lista [lim-1] Para n=lim-2 Hasta 0 con paso -1 Hacer temp= lista [lim-1] Para n=lim-2 Hasta 0 con paso -1 Hacer lista[n+1] = lista[n] FinPara FinPara FinMientras </pre>	<p> lista[n] texto </p>		

	FinPara lista[0]=temp		
<p>Imagen obtenida en pseint</p> <pre> Algoritmo Listacircular Definir lista Como Texto Definir resp Como Texto Definir temp Como Texto Definir numelem como entero Definir lim como entero Definir n como entero Dimension lista[10] // autor: Josune Singaña // fecha: 01-08-2022 para n=0 Hasta lim-1 hacer lista[n]= "" FinPara Escribir "ALGORITMO REALIZA UN DESPLAZAMIENTO CIRCULAR" lim=10 Escribir "INGRESE EL NUMERO DE ELEMENTOS DE LA LISTA " Sin Saltar Leer numelem Para n=0 hasta numelem-1 Hacer Escribir "INGRESA NOMBRE ", n+1 " " Sin Saltar Leer lista[n] FinPara resp='s' Mientras Minusculas(resp)='s' Hacer temp=lista [lim-1] Para n=lim-2 Hasta 0 con paso -1 Hacer temp= lista [lim-1] Para n=lim-2 Hasta 0 con paso -1 Hacer lista[n+1] = lista[n] FinPara FinPara lista[0]=temp Escribir " LISTA CIRCULAS ES: " Para n=0 Hasta lim-1 Hacer Escribir lista[n], " " Sin saltar FinPara Escribir " " Escribir " Realizar otro recorrido s/n" Sin Saltar leer resp FinMientras FinAlgoritmo </pre>			<p>Diseño del algoritmo</p>
<p>Imagen ejecución pseint</p>			<p>Verificación del algoritmo</p>

```
PSelnt - Ejecutando proceso LISTACIRCULAR
*** Ejecución Iniciada. ***
ALGORITMO REALIZA UN DESPLAZAMIENTO CIRCULAR
INGRESE EL NUMERO DE ELEMENTOS DE LA LISTA > 3
INGRESA NOMBRE 1 > Josu
INGRESA NOMBRE 2 > Anto
INGRESA NOMBRE 3 > Cecil
LISTA CIRCULAS ES:
Josu Anto Cecil
Realizar otro recorrido s/n> s
LISTA CIRCULAS ES:
Josu Anto Cecil
Realizar otro recorrido s/n> s
LISTA CIRCULAS ES:
Josu Anto Cecil
Realizar otro recorrido s/n> s
LISTA CIRCULAS ES:
Josu Anto Cecil
Realizar otro recorrido s/n> s
LISTA CIRCULAS ES:
Josu Anto Cecil
Realizar otro recorrido s/n> s
LISTA CIRCULAS ES:
Josu Anto Cecil
Realizar otro recorrido s/n> s
LISTA CIRCULAS ES:
Cecil Josu Anto
Realizar otro recorrido s/n> s
LISTA CIRCULAS ES:
Anto Cecil Josu
Realizar otro recorrido s/n> s
LISTA CIRCULAS ES:
Josu Anto Cecil
Realizar otro recorrido s/n> s
LISTA CIRCULAS ES:
Josu Anto Cecil
Realizar otro recorrido s/n> n
*** Ejecución Finalizada. ***
☐ No cerrar esta ventana ☐ Siempre visible 
```


Imagen obtenida en Dev c++

```
#include<stdio.h>
#include<ctype.h>
#include<string.h>
#include<conio2.h>
int main() {
    int lim,n, numelem;
    char lista[15][10];
    char resp, temp[15];
    /* autor: Josune Singaña */
    /* fecha:01-08-2022 */
    textbackground(7);
    clrscr();
    textcolor(5);
    gotoxy (25,2);printf("ALGORITMO REALIZA UN DESPLAZAMIENTO CIRCULAR\n");
    lim = 10;
    for (n=0;n<lim;n++) {
        strcpy( lista[n], "");
    }
    textcolor(1);
    gotoxy (15,4); printf("INGRESE EL NUMERO DE ELEMENTOS DE LA LISTA: ");
    scanf("%i",&numelem);
    textcolor(3);
    for (n=0;n<numelem;n++) {
        printf("INGRESA NOMBRE %i  :",n+1);
        scanf("%s",lista[n]);
    }
    resp = 's';
    while (tolower(resp)=='s') {
        strcpy(temp,lista[lim-1]);
        for (n=lim-2;n>=0;n--) {
            strcpy( temp , lista[lim-1]);
            for (n=lim-2;n>=0;n--) {
                strcpy( lista[n+1],lista[n]);
            }
        }
        strcpy( lista[0],temp);
        textcolor(12);
        printf("\n LISTA CIRCULAS ES: \n");
        for (n=0;n<lim;n++) {
            printf("%s ",lista[n]);
        }
        printf(" \n");
        textcolor(2);
        printf(" Realizar otro recorrido s/n");
        resp=getch();
    }
    getch();
    return 0;
}
```

Diseño del
algoritmo

Imagen ejecución Dev C++

Verificación
del
algoritmo

```
C:\Users\Home\Documents\1SEMESTRE\PERIODO 2022 A\PROGRAMACION 1\ALGORITMOS\DEBERES\Deb...
ALGORITMO REALIZA UN DESPLAZAMIENTO CIRCULAR

INGRESE EL NUMERO DE ELEMENTOS DE LA LISTA: 4
INGRESA NOMBRE 1 :Josu
INGRESA NOMBRE 2 :Ana
INGRESA NOMBRE 3 :Raquel
INGRESA NOMBRE 4 :Susan

LISTA CIRCULAS ES:
Josu Ana Raquel Susan
Realizar otro recorrido s/n
LISTA CIRCULAS ES:
Josu Ana Raquel Susan
Realizar otro recorrido s/n
LISTA CIRCULAS ES:
Josu Ana Raquel Susan
Realizar otro recorrido s/n
LISTA CIRCULAS ES:
Josu Ana Raquel Susan
Realizar otro recorrido s/n
LISTA CIRCULAS ES:
Josu Ana Raquel Susan
Realizar otro recorrido s/n
LISTA CIRCULAS ES:
Susan Josu Ana Raquel
Realizar otro recorrido s/n
LISTA CIRCULAS ES:
Raquel Susan Josu Ana
Realizar otro recorrido s/n
LISTA CIRCULAS ES:
Ana Raquel Susan Josu
Realizar otro recorrido s/n
LISTA CIRCULAS ES:
Josu Ana Raquel Susan
Realizar otro recorrido s/n
LISTA CIRCULAS ES:
Josu Ana Raquel Susan
Realizar otro recorrido s/n
-----
Process exited after 37.71 seconds with return value 0
```

Problema 6

11

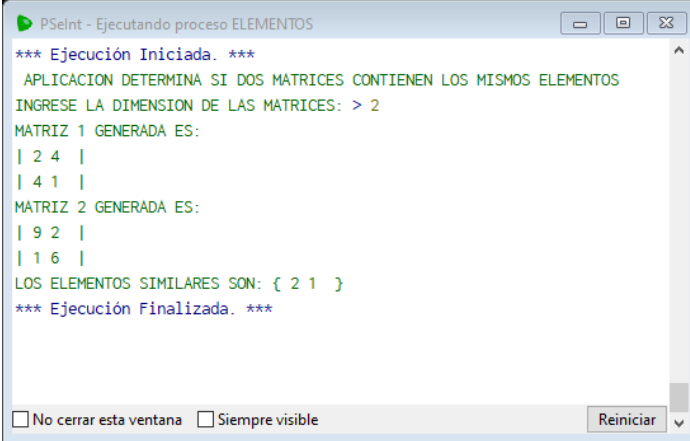
Desarrollo			Etapa
Diseñar un algoritmo para determinar si dos matrices contienen los mismos elementos, aunque estos no se presenten en el mismo orden.			Definición del problema
Entrada	Proceso	Salida	Análisis del problema
Dim E entero	Para x=1 Hasta dim Hacer para p=1 hasta dim Hacer h=mat[x,p] Para i=1 Hasta dim Hacer para j=1 hasta dim Hacer Si h = mat2[j,i] Entonces num[x]= h FinSi FinPara FinPara FinPara	p E entero	

Imagen obtenida en pseint

```
// autor: Josune Singaña
// fecha: 31-07-2022
Algoritmo elementos
    Definir i Como Entero
    Definir j Como Entero
    Definir h Como Entero
    Definir num Como Entero
    Definir mat Como Entero
    Dimension mat[100,100]
    Definir mat2 Como Entero
    Dimension mat2[100,100]
    Dimension num[100]
    Definir x Como Entero
    Definir p Como Entero
    Definir dim Como Entero
    Escribir " APLICACION DETERMINA SI DOS MATRICES CONTIENEN LOS MISMOS ELEMENTOS"
    Escribir "INGRESE LA DIMENSION DE LAS MATRICES: " Sin Saltar
    Leer dim
    para i=1 hasta dim Hacer
        para j=1 Hasta dim Hacer
            mat[i,j]=azar[9]+1
        FinPara
    FinPara
    Escribir "MATRIZ 1 GENERADA ES: "
    Para i=1 Hasta dim Hacer
        Escribir "| " Sin Saltar
        para j=1 hasta dim Hacer
            Escribir mat[i,j], " Sin Saltar
        FinPara
        Escribir " |"
    FinPara
    para i=1 hasta dim Hacer
        para j=1 Hasta dim Hacer
            mat2[i,j]=azar[9]+1
        FinPara
    FinPara
    Escribir "MATRIZ 2 GENERADA ES: "
    Para i=1 Hasta dim Hacer
        Escribir "| " Sin Saltar
        para j=1 hasta dim Hacer
            Escribir mat2[i,j], " Sin Saltar
        FinPara
        Escribir " |"
    FinPara
    Para x=1 Hasta dim Hacer
        para p=1 hasta dim Hacer
            h=mat[x,p]
            Para i=1 Hasta dim Hacer
                para j=1 hasta dim Hacer
                    Si h = mat2[j,i] Entonces
                        num[x]= h
                    FinSi
            FinPara
        FinPara
    FinPara
    x=dim*dim
    Escribir "LOS ELEMENTOS SIMILARES SON: { " Sin Saltar
    Para i=1 hasta x Hacer
        Si num[i] ≠ 0 Entonces
            Escribir num[i] " " Sin Saltar
        FinSi
    FinPara
    Escribir "}"
FinAlgoritmo
```

Diseño del
algoritmo

Imagen ejecución pseint



```
*** Ejecución Iniciada. ***
APLICACION DETERMINA SI DOS MATRICES CONTIENEN LOS MISMOS ELEMENTOS
INGRESE LA DIMENSION DE LAS MATRICES: > 2
MATRIZ 1 GENERADA ES:
| 2 4 |
| 4 1 |
MATRIZ 2 GENERADA ES:
| 9 2 |
| 1 6 |
LOS ELEMENTOS SIMILARES SON: { 2 1 }
*** Ejecución Finalizada. ***
```

**Verificación del
algoritmo**

Imagen obtenida en Dev c++

**Diseño del
algoritmo**

```

#include<stdio.h>
#include<stdlib.h>
#include<conio2.h>
/* autor: Josune Singaña */
/* fecha:31-07-2022 */
int main() {
    int dim, h, i, j, mat[100][100], mat2[100][100], num[100], p, x;
    textbackground(1);
    clrscr();
    textcolor(15);
    gotoxy (25,2);printf(" APLICACION DETERMINA SI DOS MATRICES CONTIENEN LOS MISMOS ELEMENTOS\n");
    textcolor(11);
    gotoxy (15,4);printf("INGRESE LA DIMENSION DE LAS MATRICES: ");
    scanf("%i",&dim);
    for (i=1;i<=dim;i+=1) {
        for (j=1;j<=dim;j++) {
            mat[i][j] = (rand())%9+1;
        }
    }
    textcolor(7);
    printf("MATRIZ 1 GENERADA ES: \n");
    for (i=1;i<=dim;i++) {
        printf("| ");
        for (j=1;j<=dim;j++) {
            printf("%i ",mat[i][j]);
        }
        printf(" |\n");
    }
    for (i=1;i<=dim;i++) {
        for (j=1;j<=dim;j++) {
            mat2[i][j] = (rand())%9+1;
        }
    }
    textcolor(14);
    printf("MATRIZ 2 GENERADA ES: \n");
    for (i=1;i<=dim;i++) {
        printf("| ");
        for (j=1;j<=dim;j++) {
            printf("%i ",mat2[i][j]);
        }
        printf(" |\n");
    }
    for (x=1;x<=dim;x++) {
        for (p=1;p<=dim;p++) {
            h = mat[x][p];
            for (i=1;i<=dim;i++) {
                for (j=1;j<=dim;j++) {
                    if (h==mat2[j][i]) {
                        num[x] = h;
                    }
                }
            }
        }
    }
    p=dim*dim;
    textcolor(15);
    printf("LOS ELEMENTOS SIMILARES SON: { ");
    for (i=1;i<=p;i++) {
        if (num[i]!=0) {
            printf("%i ",num[i]);
        }
    }
    printf(" }\n");
    getch();
    return 0;
}

```

Imagen ejecución Dev C++

```

C:\Users\Home\Documents\1SEMESTRE\PERIODO 2022 A\PROGRAMACION 1\ALGORITMOS\DEBERES\Deber 7\ejercicio C\elementos.exe
APLICACION DETERMINA SI DOS MATRICES CONTIENEN LOS MISMOS ELEMENTOS
INGRESE LA DIMENSION DE LAS MATRICES: 2
MATRIZ 1 GENERADA ES:
| 6 9 |
| 8 5 |
MATRIZ 2 GENERADA ES:
| 9 2 |
| 4 1 |
LOS ELEMENTOS SIMILARES SON: { 9 }
-----
Process exited after 7.97 seconds with return value 0
Presione una tecla para continuar . . .

```

**Verificación
del
algoritmo**

Problema 7

13

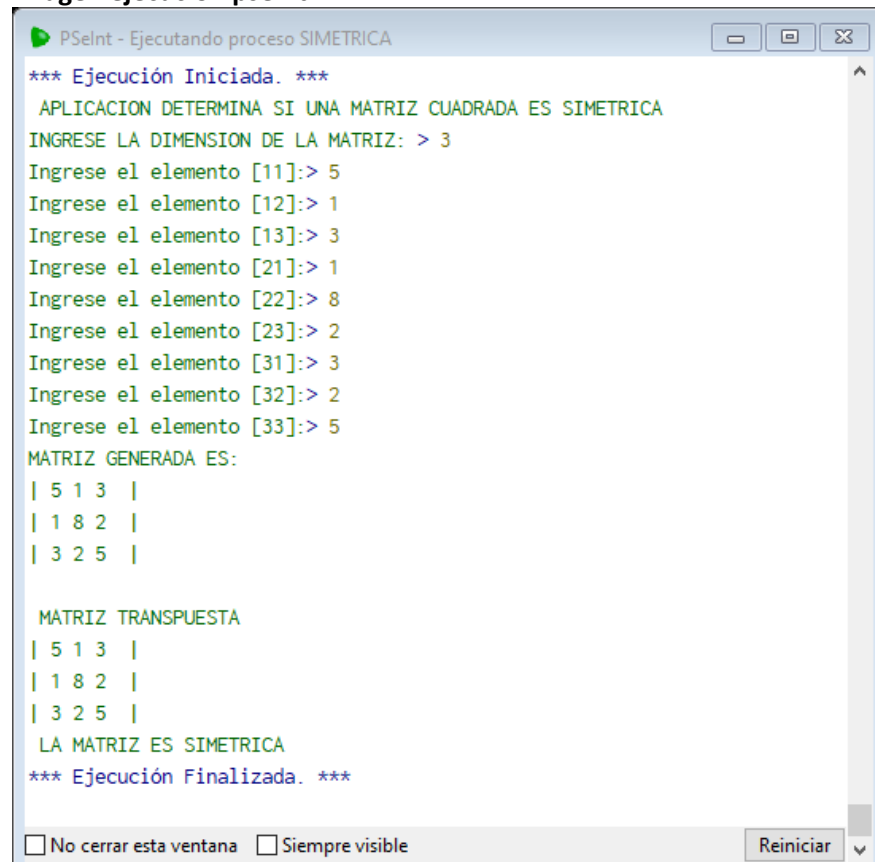
Desarrollo			Etapa
Diseñar un algoritmo para determinar si una matriz es simétrica			Definición del problema
Entrada	Proceso	Salida	Análisis del problema
Dim, mat mat[i,j] E entero	Para i=1 Hasta dim Hacer para j=1 hasta dim Hacer h=mat[i,j] Si h = mat[j,i] Entonces ele=ele+1 FinSi FinPara FinPara Si ele = tele Entonces Escribir " LA MATRIZ ES SIMETRICA " SiNo Escribir " LA MATRIZ NO ES SIMETRICA " FinSi	Matriz simétrica o no simétrica	
Imagen obtenida en pseint			Diseño del algoritmo
<pre>// autor: Josune Singaña //fecha:31-07-2022 Algoritmo simetrica definir i Como Entero Definir j Como Entero Definir h Como Entero Definir tele Como Entero Definir mat Como Entero Dimension mat[100,100] Definir ele Como Entero Definir dim Como Entero Escribir " APLICACION DETERMINA SI UNA MATRIZ CUADRADA ES SIMETRICA" Escribir "INGRESE LA DIMENSION DE LA MATRIZ: " Sin Saltar Leer dim tele=dim*dim para i=1 hasta dim Hacer para j=1 Hasta dim Hacer Escribir "Ingrese el elemento [", i,j "]: "Sin Saltar Leer mat[i,j] FinPara FinPara Escribir "MATRIZ GENERADA ES: " Para i=1 Hasta dim Hacer Escribir " " Sin Saltar para j=1 hasta dim Hacer Escribir mat[i,j], " "Sin Saltar FinPara Escribir " " FinPara</pre>			

```

Escribir ""
Escribir " MATRIZ TRANSPUESTA"
Para j=1 hasta dim Hacer
    Escribir "| " Sin Saltar
    para i=1 Hasta dim Hacer
        Escribir mat[i,j], " Sin Saltar
    FinPara
    Escribir " |"
FinPara
Para i=1 Hasta dim Hacer
    para j=1 hasta dim Hacer
        h=mat[i,j]
        Si h = mat[j,i] Entonces
            ele=ele+1
        FinSi
    FinPara
FinPara
Si ele = tele Entonces
    Escribir " LA MATRIZ ES SIMETRICA "
SiNo
    Escribir " LA MATRIZ NO ES SIMETRICA "
FinSi
FinAlgoritmo

```

Imagen ejecución pseint



**Verificación del
algoritmo**

Imagen obtenida en Dev c++

```
#include<conio2.h>
#include<stdio.h>
/* autor: Josune Singaña */
/* fecha:31-07-2022 */
int main() {
    int dim, ele, h, i, j, mat[100][100], tele;
    ele=0;
    textbackground(3);
    clrscr();
    textcolor(1);
    gotoxy (25,2);printf(" APLICACION DETERMINA SI UNA MATRIZ CUADRADA ES SIMETRICA\n");
    textcolor(5);
    gotoxy (15,4);printf("INGRESE LA DIMENSION DE LA MATRIZ: ");
    scanf("%i",&dim);
    tele = dim*dim;
    textcolor(7);
    for (i=1;i<=dim;i+=1) {
        for (j=1;j<=dim;j+=1) {
            printf("Ingrese el elemento [%i%i]:",i,j);
            scanf("%i",&mat[i][j]);
        }
    }
    textcolor(12);
    printf("MATRIZ GENERADA ES: \n");
    for (i=1;i<=dim;i+=1) {
        printf("| ");
        for (j=1;j<=dim;j+=1) {
            printf("%i ",mat[i][j]);
        }
        printf(" |\n");
    }
    printf("\n");
    textcolor(13);
    printf(" MATRIZ TRANSPUESTA\n");
    for (j=1;j<=dim;j+=1) {
        printf("| ");
        for (i=1;i<=dim;i+=1) {
            printf("%i ",mat[i][j]);
        }
        printf(" |\n");
    }
    for (i=1;i<=dim;i+=1) {
        for (j=1;j<=dim;j+=1) {
            h = mat[i][j];
            if (h==mat[j][i]) {
                ele = ele+1;
            }
        }
    }
    textcolor(1);
    if (ele==tele) {
        printf(" LA MATRIZ ES SIMETRICA \n");
    } else {
        printf(" LA MATRIZ NO ES SIMETRICA \n");
    }
    getch();
    return 0;
}
```

Diseño del
algoritmo

Imagen ejecución Dev C++

Verificación
del
algoritmo

```

C:\Users\Home\Documents\1SEMESTRE\PERIODO 2022 A\PROGRAMACION 1\ALGORITMOS\DEBERES\Deber 7\ejercicio 1
APLICACION DETERMINA SI UNA MATRIZ CUADRADA ES SIMETRICA

INGRESE LA DIMENSION DE LA MATRIZ: 3
Ingrese el elemento [11]:5
Ingrese el elemento [12]:1
Ingrese el elemento [13]:3
Ingrese el elemento [21]:1
Ingrese el elemento [22]:8
Ingrese el elemento [23]:2
Ingrese el elemento [31]:3
Ingrese el elemento [32]:2
Ingrese el elemento [33]:5
MATRIZ GENERADA ES:
| 5 1 3 |
| 1 8 2 |
| 3 2 5 |

MATRIZ TRANSPUESTA
| 5 1 3 |
| 1 8 2 |
| 3 2 5 |
LA MATRIZ ES SIMETRICA

-----
Process exited after 35.04 seconds with return value 0
Presione una tecla para continuar . . .

```

Problema 8

15

Desarrollo				Etapa
Diseñar un algoritmo para calcular el producto de dos matrices				Definición del problema
Entrada Nfa, nca, nfb, ncb E enteros	Proceso Si nfr= nc2 Entonces nfr=nf1 ncr=nc2 para i=0 hasta nfr-1 hacer para j=0 hasta ncr-1 hacer mr[i,j]=0 para k=0 hasta nc1-1 hacer mr[i,j]=mr[i,j]+m1[i,k]*m2[k,j] FinPara FinPara FinPara SiNo Escribir "*****MULTIPLICACIÓN MATRICIAL NO DEFINIDA*****" FinSi	Salida m[i,j], E entero		Análisis del problema

Imagen obtenida en pseint

```
SubProceso generarMatriz(nf,nc,m)
    definir i Como Entero
    definir j Como Entero
    para i=0 hasta nf-1 hacer
        para j=0 hasta nc-1 hacer
            m[i,j]=azar(9)+1
        FinPara
    FinPara
FinSubProceso

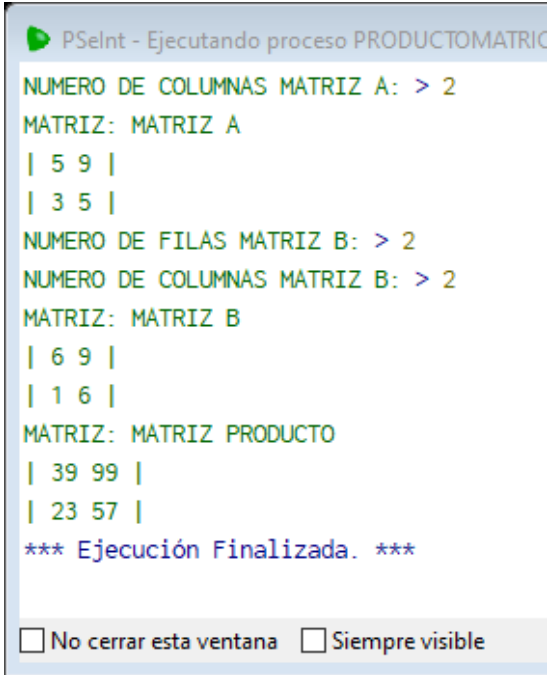
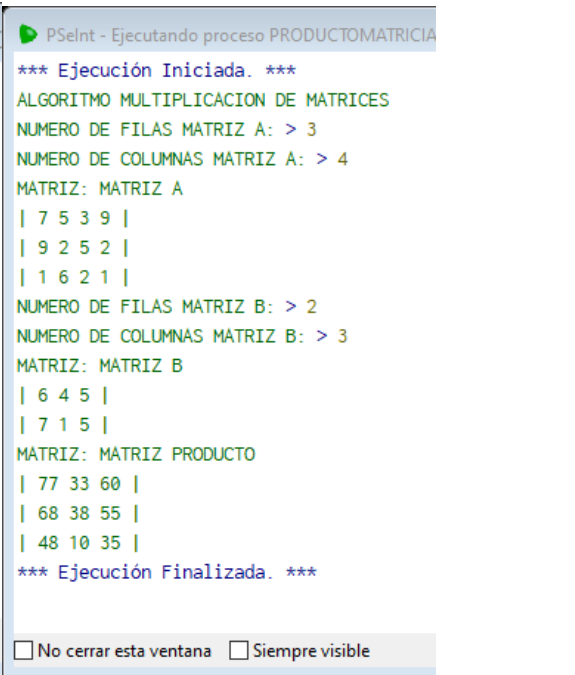
SubProceso productomatriz(nf1,nc1,m1, nf2,nc2,m2, nfr Por Referencia,ncr Por Referencia, mr)
    definir i Como Entero
    definir j Como Entero
    definir k Como Entero
    Si nf1= nc2 Entonces
        nfr=nf1
        ncr=nc2
        para i=0 hasta nfr-1 hacer
            para j=0 hasta ncr-1 hacer
                mr[i,j]=0
                para k=0 hasta nc1-1 hacer
                    mr[i,j]=mr[i,j]+m1[i,k]*m2[k,j]
                FinPara
            FinPara
        FinPara
    SiNo
        Escribir "*****MULTIPLICACIÓN MATRICIAL NO DEFINIDA*****"
    FinSi
FinSubProceso

SubProceso escribirMatriz(nf,nc,m,nommat)
    definir i Como Entero
    definir j Como Entero
    Escribir "MATRIZ: ", nommat
    para i=0 hasta nf-1 hacer
        Escribir "| " Sin Saltar
        para j=0 hasta nc-1 hacer
            Escribir m[i,j], " " Sin Saltar
        FinPara
        Escribir "| "
    FinPara
FinSubProceso

Algoritmo Productomatricial
    Definir ma Como Entero
    Definir mb Como Entero
    Definir mc Como Entero
    Definir i Como Entero
    Definir j Como Entero
    Definir k Como Entero
    Definir nfa Como Entero
    Definir nca Como Entero
    Definir nfb Como Entero
    Definir ncb Como Entero
    Definir nfc Como Entero
    Definir ncc Como Entero
    Dimension ma[10,10]
    Dimension mb[10,10]
```

Diseño del
algoritmo

<pre> Dimension mc[10,10] // autor: Josune Singaña //fecha:01-08-2022 Escribir "ALGORITMO MULTIPLICACION DE MATRICES " Escribir "NUMERO DE FILAS MATRIZ A: "Sin Saltar Leer nfa Escribir "NUMERO DE COLUMNAS MATRIZ A: "Sin Saltar Leer nca generarMatriz(nfa,nca,ma) escribirmatriz(nfa,nca,ma,"MATRIZ A") Escribir "NUMERO DE FILAS MATRIZ B: "Sin Saltar Leer nfb Escribir "NUMERO DE COLUMNAS MATRIZ B: "Sin Saltar Leer ncb generarMatriz(nfb,ncb,mb) escribirmatriz(nfb,ncb,mb,"MATRIZ B") productomatriz(nfa, nca, ma, nfb, ncb,mb, nfc, ncc, mc) escribirmatriz(nfc,ncc,mc,"MATRIZ PRODUCTO") FinAlgoritmo </pre>	
---	--

<p>Imagen ejecución pseint</p> <div data-bbox="178 953 721 1623">  <pre> PSeInt - Ejecutando proceso PRODUCTOMATRIZ NUMERO DE COLUMNAS MATRIZ A: > 2 MATRIZ: MATRIZ A 5 9 3 5 NUMERO DE FILAS MATRIZ B: > 2 NUMERO DE COLUMNAS MATRIZ B: > 3 MATRIZ: MATRIZ B 6 9 1 6 MATRIZ: MATRIZ PRODUCTO 39 99 23 57 *** Ejecución Finalizada. *** <input type="checkbox"/> No cerrar esta ventana <input type="checkbox"/> Siempre visible </pre> </div> <div data-bbox="721 953 1292 1623">  <pre> PSeInt - Ejecutando proceso PRODUCTOMATRIZ *** Ejecución Iniciada. *** ALGORITMO MULTIPLICACION DE MATRICES NUMERO DE FILAS MATRIZ A: > 3 NUMERO DE COLUMNAS MATRIZ A: > 4 MATRIZ: MATRIZ A 7 5 3 9 9 2 5 2 1 6 2 1 NUMERO DE FILAS MATRIZ B: > 2 NUMERO DE COLUMNAS MATRIZ B: > 3 MATRIZ: MATRIZ B 6 4 5 7 1 5 MATRIZ: MATRIZ PRODUCTO 77 33 60 68 38 55 48 10 35 *** Ejecución Finalizada. *** <input type="checkbox"/> No cerrar esta ventana <input type="checkbox"/> Siempre visible </pre> </div>	<p>Verificación del algoritmo</p>
---	-----------------------------------

<p>Imagen obtenida en Dev c++</p>	<p>Diseño del algoritmo</p>
-----------------------------------	-----------------------------

```

#include<stdio.h>
#include<stdlib.h>
#include<conio2.h>
void generarmatriz(int nf, int nc, int m[10][10]);
void productomatriz(int nf1, int nc1, int m1[10][10], int nf2, int nc2, int m2[10][10], int *nfr, int *ncr, int mr[20][20]);
void escribirmatriz(int nf, int nc, int m[10][10], char nommat);

void generarmatriz(int nf, int nc, int m[10][10]) {
    int i, j;
    for (i=0; i<nf-1; i++) {
        for (j=0; j<nc-1; j++) {
            m[i][j] = (rand()%9)+1;
        }
    }
}

void productomatriz(int nf1, int nc1, int m1[10][10], int nf2, int nc2, int m2[10][10], int *nfr, int *ncr, int mr[20][20]) {
    int i, j, k;
    if (nf1==nc2) {
        (*nfr) = nf1;
        (*ncr) = nc2;
        for (i=0; i<(*nfr)-1; i++) {
            for (j=0; j<(*ncr)-1; j++) {
                mr[i][j] = 0;
                for (k=0; k<nc1-1; k++) {
                    mr[i][j] = mr[i][j]+m1[i][k]*m2[k][j];
                }
            }
        }
    } else {
        printf("*****MULTIPLICACIÓN MATRICIAL NO DEFINIDA*****\n");
    }
}

void escribirmatriz(int nf, int nc, int m[10][10], char nommat) {
    int i, j;
    printf("MATRIZ: %i\n", nommat);
    for (i=0; i<nf-1; i++) {
        printf("| ");
        for (j=0; j<nc-1; j++) {
            printf("%i ", m[i][j]);
        }
        printf("| \n");
    }
}

int main() {
    int i, j, k, nca, ncb, ncc, nfa, nfb, nfc;
    int ma[10][10], mb[10][10], mc[20][20];
    /* autor: Josune Singaña */
    /* fecha: 01-08-2022 */
    textbackground(0);
    clrscr();
    textcolor(12);
    gotoxy(25,2); printf("ALGORITMO MULTIPLICACION DE MATRICES \n");
    textcolor(14);
    printf("NUMERO DE FILAS MATRIZ A: ");
    scanf("%i", &nfa);
    printf("NUMERO DE COLUMNAS MATRIZ A: ");
    scanf("%i", &nca);
    generarmatriz(nfa, nca, ma);
    escribirmatriz(nfa, nca, ma, "MATRIZ A");
    textcolor(11);
    printf("NUMERO DE FILAS MATRIZ B: ");
    scanf("%i", &nfb);
    printf("NUMERO DE COLUMNAS MATRIZ B: ");
    scanf("%i", &ncb);
    generarmatriz(nfb, ncb, mb);
    escribirmatriz(nfb, ncb, mb, "MATRIZ B");
    productomatriz(nfa, nca, ma, nfb, ncb, mb, &nfc, &ncc, mc);
    escribirmatriz(nfc, ncc, mc, "MATRIZ PRODUCTO");
    getch();
    return 0;
}

```

Imagen ejecución Dev C++

```

ALGORITMO MULTIPLICACION DE MATRICES

NUMERO DE FILAS MATRIZ A: 2
NUMERO DE COLUMNAS MATRIZ A: 2
MATRIZ: -90
| 6 9 |
| 8 5 |
NUMERO DE FILAS MATRIZ B: 2
NUMERO DE COLUMNAS MATRIZ B: 2
MATRIZ: -24
| 9 2 |
| 4 1 |

-----
Process exited after 13.87 seconds with return value 3221225477
Presione una tecla para continuar . . .

```

Verificación
del
algoritmo

```

C:\Users\Home\Documents\1SEMESTRE\PERIODO 2022 A\PROGRAMACION 1\ALGORITMOS\DEBERES

ALGORITMO MULTIPLICACION DE MATRICES
NUMERO DE FILAS MATRIZ A: 3
NUMERO DE COLUMNAS MATRIZ A: 2
MATRIZ: -90
| 6 9 |
| 8 5 |
| 9 2 |
NUMERO DE FILAS MATRIZ B: 2
NUMERO DE COLUMNAS MATRIZ B: 3
MATRIZ: -24
| 4 1 8 |
| 3 9 3 |

-----
Process exited after 8.18 seconds with return value 3221225477
Presione una tecla para continuar . . .

```

Problema 9

17

Desarrollo			Etapas
Para facturar el servicio de energía la empresa Energía Para Todos cuenta con una lista de usuarios almacenada en un vector y la lista de lecturas del mes anterior en un segundo vector. Se requiere un algoritmo que lea el valor del kW, tome la lectura actual de cada usuario y la registre en un tercer vector, calcule el consumo del mes por diferencia de lecturas y muestre para cada usuario: nombre, consumo y valor a pagar.			Definición del problema
Entrada	Proceso	Salida	Análisis del problema
Numus, Lecant[i], Lecact[i] E entero Istusuarios[i] E texto	Para i=0 Hasta numus-1 Hacer Escribir"NOMBRE DE USUARIO: ", Istusuarios[i] Escribir"VALOR DEL CONSUMO EN KWH: ", lecact[i] - lecant[i] Escribir"VALOR A PAGAR: ", (lecact[i] - lecant[i])*costkwh FinPara	Istusuarios[i] E texto lecact[i] - lecant[i], (lecact[i] - lecant[i])*costkwh E reales	
Imagen obtenida en pseint			Diseño del algoritmo

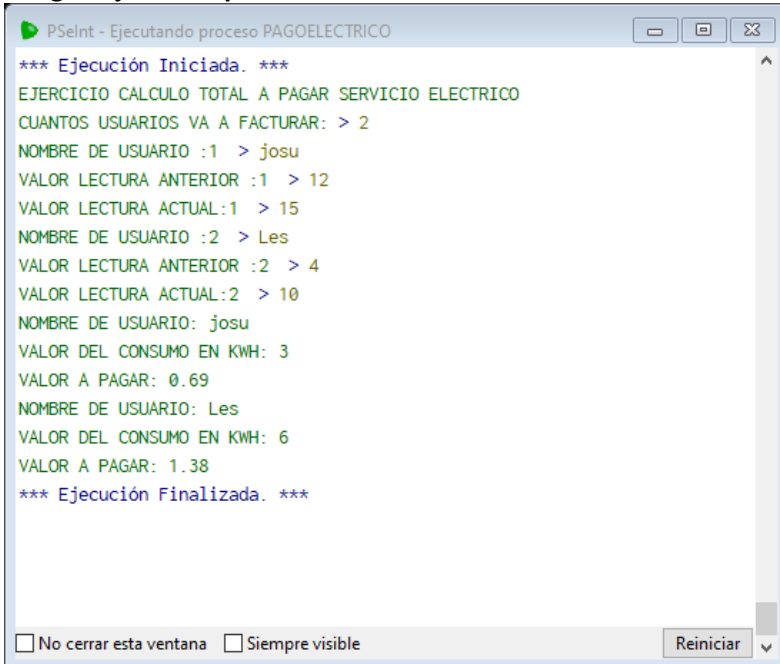
```

Algoritmo PagoElectrico
  Definir lstusuarios Como Texto
  Definir Lecant Como entero
  Definir Lecact Como entero
  Definir numus como entero
  Definir costkwh como real
  Definir i como entero
  Dimension lstusuarios[10]
  Dimension Lecant[10]
  Dimension Lecact[10]
  // autor: Josune Singaña
  // fecha: 01-08-2022
  costkwh=0.23
  Escribir "EJERCICIO CALCULO TOTAL A PAGAR SERVICIO ELECTRICO"

  Escribir "CUANTOS USUARIOS VA A FACTURAR: " Sin Saltar
  Leer numus
  Para i=0 hasta numus-1 Hacer
    Escribir "NOMBRE DE USUARIO :", i+1 " Sin Saltar
    Leer lstusuarios[i]
    Escribir "VALOR LECTURA ANTERIOR :", i+1 " Sin Saltar
    Leer Lecant[i]
    Escribir "VALOR LECTURA ACTUAL:", i+1 " Sin Saltar
    Leer Lecact[i]
  FinPara
  // CALCULO CONSUMO Y VALOR A PAGAR
  Para i=0 Hasta numus-1 Hacer
    Escribir "NOMBRE DE USUARIO: ", lstusuarios[i]
    Escribir "VALOR DEL CONSUMO EN KWH: ", Lecact[i] - Lecant[i]
    Escribir "VALOR A PAGAR: ", (Lecact[i] - Lecant[i])*costkwh
  FinPara
FinAlgoritmo

```

Imagen ejecución pseint



```

*** Ejecución Iniciada. ***
EJERCICIO CALCULO TOTAL A PAGAR SERVICIO ELECTRICO
CUANTOS USUARIOS VA A FACTURAR: > 2
NOMBRE DE USUARIO :1 > josu
VALOR LECTURA ANTERIOR :1 > 12
VALOR LECTURA ACTUAL:1 > 15
NOMBRE DE USUARIO :2 > Les
VALOR LECTURA ANTERIOR :2 > 4
VALOR LECTURA ACTUAL:2 > 10
NOMBRE DE USUARIO: josu
VALOR DEL CONSUMO EN KWH: 3
VALOR A PAGAR: 0.69
NOMBRE DE USUARIO: Les
VALOR DEL CONSUMO EN KWH: 6
VALOR A PAGAR: 1.38
*** Ejecución Finalizada. ***

```

Verificación del algoritmo

Imagen obtenida en Dev c++

```
#include<stdio.h>
#include<conio2.h>
int main() {
    float costkwh;
    int i, lecact[10], lecant[10], numus;
    char lstusuarios[40][40];
    /* autor: Josune Singaña */
    /* fecha:01-08-2022 */
    costkwh = 0.23;
    textbackground(14);
    clrscr();
    textcolor(12);
    gotoxy(25,2);printf("EJERCICIO CALCULO TOTAL A PAGAR SERVICIO ELECTRICO\n");
    textcolor(1);
    gotoxy(15,4); printf("CUANTOS USUARIOS VA A FACTURAR: ");
    scanf("%i",&numus);
    for (i=1;i<=numus;i++) {
        lecact[i]=0;
        lecant[i]=0;
    }
    textcolor(0);
    for (i=1;i<=numus;i++) {
        printf("\nNOMBRE DE USUARIO %i : ",i);
        scanf("%s",lstusuarios[i]);
        printf("VALOR LECTURA ANTERIOR %i : ",i);
        scanf("%i",&lecant[i]);
        printf("VALOR LECTURA ACTUAL%i : ",i);
        scanf("%i",&lecact[i]);
    }
    /* CALCULO CONSUMO Y VALOR A PAGAR */
    textcolor(4);
    for (i=1;i<=numus;i++) {
        printf("\nNOMBRE DE USUARIO: %s\n",lstusuarios[i]);
        printf("VALOR DEL CONSUMO EN KWH: %i\n",lecact[i]-lecant[i]);
        printf("VALOR A PAGAR: %.2f\n",(lecact[i]-lecant[i])*costkwh);
    }
    getch();
    return 0;
}
```

Diseño del
algoritmo

Imagen ejecución Dev C++

```
C:\Users\Home\Documents\1SEMESTRE\PERIODO 2022 A\PROGRAMACION 1\ALGORITMOS\DEBERES\Deber 7\ej...

EJERCICIO CALCULO TOTAL A PAGAR SERVICIO ELECTRICO

CUANTOS USUARIOS VA A FACTURAR: 2

NOMBRE DE USUARIO 1 : Josu
VALOR LECTURA ANTERIOR 1 : 12
VALOR LECTURA ACTUAL1 : 15

NOMBRE DE USUARIO 2 : Les
VALOR LECTURA ANTERIOR 2 : 4
VALOR LECTURA ACTUAL2 : 10

NOMBRE DE USUARIO: Josu
VALOR DEL CONSUMO EN KWH: 3
VALOR A PAGAR: 0.69

NOMBRE DE USUARIO: Les
VALOR DEL CONSUMO EN KWH: 6
VALOR A PAGAR: 1.38

-----
Process exited after 18.81 seconds with return value 0
Presione una tecla para continuar . . .
```

Verificación
del
algoritmo

