



# Escuela Politécnica Nacional



# Facultad De Ingeniería en Sistemas PROGRAMACIÓN I (COMPONENTE PRÁCTICO)

**DOCENTE:** Eddie Hans Yánez Quezada

ESTUDIANTE: Josune Antonella Singaña Tapia

**TEMA:** Recursividad

**FECHA:** 24 de julio de 2022

**INDICE** 

**Objetivos** 

Problema 1

Problema 2

Problema 3

Problema 4

Problema 5

Problema 6

# **OBJETIVOS DEL DEBER:**

- Comprender el uso de recursividad para la resolución de ejercicios.
- Analizar la estructura en lenguaje C del algoritmo desarrollado en pseint.
- Resolver los ejercicios pares de la Página 375 del libro de Chaves

# ACTIVIDAD EJERCICIO DE ALGORITMOS

# Problema 1

2

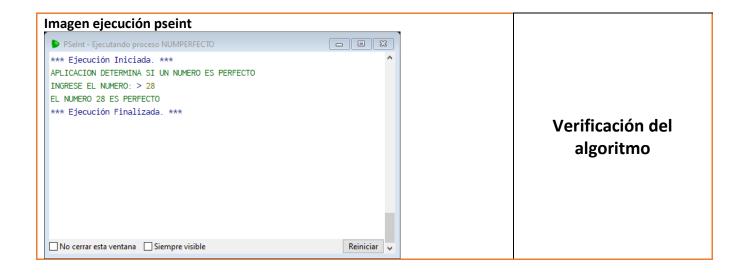
	Des	sarrollo		Etapa
Generar la tal	la de multiplicar de un	número		Definición del problema
Entrada	Proceso		Salida	
num E real.	· ·	producto(n,m-1) bir n " x ",m " =", p	p E reales	Análisis del problema
// autor: Jos //fecha:24-07 Funcion p=pr definir p Si m>0 E p=n+p Escri FinSi FinFuncion Algoritmo ta Definir p Escribir	CULA E IMPRIME  June Singaña  June Singaña  COULA E IMPRIME  June Singaña  COULA E IMPRIME  June Singaña  COMO real  June Singaña  June Singañ			Diseño del algoritmo
► PSeInt - Ejecutando *** Ejecución Inic APLICACION CALCULA	CULA E IMPRIME  Proceso TABLAMULTIPLICACION  Lada. ***  LA TABLA DE MULTIPLICAR DE UN NU  A CALCULAR LA TABLA: > 5.1	JMERO SS		Verificación del algoritmo

```
Imagen obtenida en Dev c++
#include<stdio.h>
#include<conio2.h>
int producto(int n, int m);
int producto(int n, int m) {
    if (m>0) {
       producto(n,m-1);
printf ("\t\t %d x %d =%d\n",n,m,(n*m));
                                                                                                   Diseño del
                                                                                                   algoritmo
int main() {
    textbackground(YELLOW);
    clrscr();
textcolor(LIGHTRED);
    int num;
gotoxy (10,2);printf("APLICACION CALCULA LA TABLA DE MULTIPLICAR DE UN NUMERO \n");
    textcolor(LIGHTBLUE);
gotoxy (15,4);printf("INGRESE EL NUMERO A CALCULAR LA TABLA: ");
scanf("%d",&num);
producto(num,12);
                                Imagen ejecución Dev C++
     C:\Users\Home\Documents\1SEMESTRE\PERIODO 2022 A\PROGRAMACION 1\ALGORITMOS\DEB
                APLICACION CALCULA LA TABLA DE MULTIPLICAR DE UN NUMERO
                       INGRESE EL NUMERO A CALCULAR LA TABLA: 7
                          7 x 1 =7
                          7 \times 2 = 14
                          7 \times 3 = 21
                                                                                                  Verificación
                          7 \times 4 = 28
                          7 \times 5 = 35
                                                                                                        del
                          7 \times 6 = 42
                                                                                                   algoritmo
                          7 x 7 =49
                          7 \times 8 = 56
                          7 \times 9 = 63
                          7 x 10 = 70
                         7 x 11 =77
                          7 x 12 =84
    Process exited after 1.525 seconds with return value 0
    Presione una tecla para continuar . . .
```

# Problema 2



	Desarrollo		Etapa
Determinar si u	n número es perfecto		Definición del problema
Entrada Num E enteros	Proceso  Si m > 0 Entonces sd=divisor(n,m-1) Si n<>m y n mod m=0 Entonces sd=sd+m FinSi FinSi	Salida pe E entero	Análisis del problema
//fecha:24- Funcion sd Defini Si m > sd= Si Fin FinSi FinFuncion Funcion d= defini d=divis FinFuncion Algoritmo Defini Escrib Escrib Leer no s=recur Escrib Si s==n Escrib Si s==n	### Singaña  ### O7-2022  ### Como Entero  ### DENTONCES  ### DENTONCES  ### DENTONCES  ### DENTONCES  #### DENTONCES  #### DENTONCES  #### DENTONCES  #### DENTONCES  ###################################		Diseño del algoritmo



```
Imagen obtenida en Dev c++
#include<conio2.h>
long divisor(int n, int m, long sd);
long recursum(int n);
/* autor: Josune Singaña */
/* fecha:24-07-2022 */
long divisor(int n, int m,long sd) {
    if (m>0)
        sd = divisor(n,m-1,sd);
        if (n!=m && n%m ==0) {
    sd = sd+m;
    return sd;
long recursum(int n) {
int d;
    d = divisor(n,n,0);
    return d;
                                                                                              Diseño del
int main() {
                                                                                              algoritmo
    textbackground(BLACK);
    clrscr();
    textcolor(LIGHTCYAN);
    int num;
    long s;
        gotoxy (40,2);printf("APLICACION DETERMINA SI UN NUMERO ES PERFECTO \n");
        textcolor(LIGHTGREEN);
    gotoxy (30,4);printf("INGRESE EL NUMERO ENTERO: ");
scanf("%i",&num);
    s = recursum(num);
        textcolor(LIGHTRED);
    gotoxy (35,6);printf("EL NUMERO %i",num);
    if (s==num) {
        printf(" ES PERFECTO \n");
        printf(" NO ES PERFECTO \n");
    getch();
return 0;
```



#### Problema 3



	Desarrollo		Etapa
Calcular el	Mínimo Común Múltiplo (MCM) de dos núm	eros. ( <b>algoritmo de Euclides</b>	Definición del problema
Entrada	Proceso	Salida	
N, N2 E enteros	Si m <>n Entonces Si n mod m = 0 Entonces mcd=m Sino mcd=maxcodivi(m,n mod m) FinSi FinSi h=maxcodivi(n,m) d=(n*m)/h	d E enteros	Análisis del problema

```
Imagen obtenida en pseint
 Funcion mcd=maxcodivi(n,m)
      Definir mcd Como Entero
      Si m ≠n Entonces
          Si n mod m = 0 Entonces
          Sino
              mcd=maxcodivi(m,n mod m)
          FinSi
      FinSi
 FinFuncion
 Funcion d=recursum(n,m)
      definir d Como Entero
                                                                                     Diseño del
      definir h Como Entero
                                                                                     algoritmo
      h=maxcodivi(n,m)
      d=(n*m)/h
 FinFuncion
 Algoritmo mincomunmult
      Definir n Como Entero
      Definir n2 Como entero
      Definir s Como entero
      Escribir "APLICACION DETERMINA EL MINIMO COMUN MULTPLO "
      Escribir "INGRESE EL PRIMER NUMERO: " Sin Saltar
      Leer n
      Escribir"INGRESE EL SEGUNDO NUMERO: " Sin Saltar
      Leer n2
      s=recursum(n,n2)
      Escribir "EL MINIMO COMUN MULTPLO ES: ", s
 FinAlgoritmo
Imagen ejecución pseint
                                            - E X
 PSeInt - Ejecutando proceso MINCOMUNMULT
*** Ejecución Iniciada. ***
APLICACION DETERMINA EL MINIMO COMUN MULTPLO
INGRESE EL PRIMER NUMERO: > 60
INGRESE EL SEGUNDO NUMERO: > 45
EL MINIMO COMUN MULTPLO ES:180
*** Ejecución Finalizada. ***
                                                                                  Verificación del
                                                                                      algoritmo
                                               Reiniciar 🗸
☐ No cerrar esta ventana ☐ Siempre visible
```

```
Imagen obtenida en Dev c++
#include<stdio.h>
#include<conio2.h>
// Declaraciones adelantadas de las funciones
int maxcodivi(int n, int m);
int recursum(int n, int m);
int maxcodivi(int n, int m) {
    int mcd;
    if (m!=n) {
        if (n%m==0) {
            mcd = m;
            mcd = maxcodivi(m,n%m);
    return mcd;
int recursum(int n, int m) {
                                                                                        Diseño del
    int d;
                                                                                         algoritmo
    int h;
    h = maxcodivi(n,m);
    d = (n*m)/h;
    return d;
int main() {
    int n, n2, s;
       textbackground(BLUE);
    clrscr();
    textcolor(LIGHTCYAN);
        gotoxy (25,2);printf("APLICACION DETERMINA EL MINIMO COMUN MULTPLO \n");
            textcolor(WHITE);
    printf("\n\tINGRESE EL PRIMER NUMERO: ");
    scanf("%i",&n);
    printf("\n\tINGRESE EL SEGUNDO NUMERO: ");
    scanf("%i",&n2);
    s = recursum(n,n2);
    printf("\n\tEL MINIMO COMUN MULTPLO ES:%i\n",s);
    getch();
    return 0;
                             Imagen ejecución Dev C++
 ■ C:\Users\Home\Documents\1SEMESTRE\PERIODO 2022 A\PROGRAMACION 1\ALGORITMOS\DEBERES\DEBER 6\ejercicios en C\min
                       APLICACION DETERMINA EL MINIMO COMUN MULTPLO
                                                                                       Verificación
        INGRESE EL PRIMER NUMERO: 4
                                                                                             del
        INGRESE EL SEGUNDO NUMERO: 3
                                                                                         algoritmo
        EL MINIMO COMUN MULTPLO ES:12
 Process exited after 2.139 seconds with return value 0
  resione una tecla para continuar . . .
```

	Desarrollo		Etapa
Convertir un número	binario a decimal		Definición del problema
Entrada	Proceso	Salida	
num E enteros	Si n <> 0 Entonces bi= ConvertirAtexto(n%2)+bi n=trunc(n /2) escribir bi Sin Saltar bi=binario(n) FinSi	bi E Caracter.	Análisis del problema
Imagen obtenida e			
// autor: Josum //fecha:24-07-2 SubProceso bi=     Definir bi     Definir m (     Si n ≠ 0 En     bi= Convert     n=trunc(n //     escribir b     bi=binario() FinSi FinSubProceso  Algoritmo deci     Definir num     Escribir "     Escribir "     Leer num	e Singaña  022  binario(n)  Como Caracter  Como real  tonces  irAtexto(n%2)+bi  2)  i Sin Saltar  n)	NVERTIR: " Sin Saltar	Diseño del algoritmo
Imagen ejecución p	seint	=	
▶ PSeInt - Ejecutando proceso DE *** Ejecución Iniciada. ** APLICACION CONVIERTE NUMEI INGRESE EL NUMERO QUE DESE. 12 EN BINARIO ES:0011 *** Ejecución Finalizada. :  □ No cerrar esta ventana □ Siem	* RO DECIMAL EN BINARIO A CONVERTIR: > 12 ****		Verificación del algoritmo

#### Imagen obtenida en Dev c++ Forma 1: fecha: 24 de Julio de 2022\*/ #include<stdio.h> #include<conio2.h> void binario(int n) if (n!=0) binario(n/2); printf("%i",n%2); int main() int num=0; textbackground(MAGENTA); clrscr(); textcolor(LIGHTGRAY); gotoxy (30,2);printf("APLICACION CONVIERTE NUMERO DECIMAL EN BINARIO\n"); textcolor(WHITE); gotoxy (25,4);printf("INGRESE EL NUMERO QUE DESEA CONVERTIR: "); scanf("%i",&num); //Pedir variable num textcolor(YELLOW); gotoxy (40,6); binario(num); Diseño del Forma 2: algoritmo #include<stdio.h> #include<conio2.h> /\*autor: Josune Singaña fecha: 24 de Julio de 2022\*/ long binario(int n) { long r=0; if (n>0) r=binario (n/2); r=r\*10+(n%2); return r; int main() { textbackground(LIGHTGRAY); clrscr(); textcolor(MAGENTA); int num; gotoxy (30,2);printf("APLICACION CONVIERTE NUMERO DECIMAL EN BINARIO\n"); textcolor(CYAN); gotoxy (25,4);printf("INGRESE EL NUMERO QUE DESEA CONVERTIR: "); scanf("%i",&num); //Pedir variable num textcolor(BLUE); gotoxy (40,5 ); printf("%ld\n",binario(num)); return 0;



# Problema 5

10

	Desarrollo		Etapa
Mostrar los prim	neros n términos de la serie Fibonacci		Definición del problema
Entrada	Proceso	Salida	
L, num, i, j, fibo E enteros	Si i<=l Entonces Escribir" ", n Sin Saltar fibo=n+j j=n n=fibo fibonum=serie(l,i+1,n,fibo,j) FinSi	n E enteros	Análisis del problema
lmagen obtenid	a en pseint		Diseño del algoritmo

```
Funcion fibonum=serie(l,i,n,fibo,j)
          Definir fibonum Como Entero
          Si i≤l Entonces
              Escribir" ", n Sin Saltar
              fibo=n+j
             j=n
              n=fibo
              fibonum=serie(1,i+1,n,fibo,j)
          FinSi
      FinFuncion
      Funcion d=recursum(1)
          definir d Como Entero
          d=serie(1,1,1,0,0)
     FinFuncion
     Algoritmo seriefibo
          Definir 1 Como Entero
          Definir s Como Entero
          s=0
          Escribir " APLICACION GENERA SERIE DE NUMEROS DE FIBONACCI "
          Escribir "INGRESE EL LIMITE DE LA SERIE: " Sin Saltar
         Leer 1
          Escribir "LA SERIE ES: " Sin saltar
          si l= 0 o l=1 Entonces
              Escribir " ",1 Sin saltar
          FinSi
          s=recursum(1)
          Escribir " "
 30 FinAlgoritmo
Imagen ejecución pseint
                                           PSeInt - Ejecutando proceso SERIEFIBO
*** Ejecución Iniciada. ***
APLICACION GENERA SERIE DE NUMEROS DE FIBONACCI
INGRESE EL LIMITE DE LA SERIE: > 10
LA SERIE ES: 1 1 2 3 5 8 13 21 34 55
*** Ejecución Finalizada. ***
                                                                                Verificación del
                                                                                    algoritmo
☐ No cerrar esta ventana ☐ Siempre visible
```

#### Imagen obtenida en Dev c++ #include<stdio.h> #include<conio2.h> int serie(float 1, float i, float n, float fibo, float j); int recursum(float 1); int serie(float 1, float i, float n, float fibo, float j) { int fibonum; if (i<=1) { printf(" %.0f",n);</pre> fibo = n+j; j = n;n = fibo; fibonum = serie(l,i+1,n,fibo,j); return fibonum; int recursum(float 1) { int d; Diseño del d = serie(1,1,1,0,0); return d; algoritmo int main() { textbackground(LIGHTGRAY); clrscr(); textcolor(RED); int 1, s; s = 0;gotoxy (40,2);printf(" APLICACION GENERA SERIE DE NUMEROS DE FIBONACCI \n"); textcolor(LIGHTMAGENTA); gotoxy (30,4); printf("INGRESE EL LIMITE DE LA SERIE: "); scanf("%i",&1); textcolor(GREEN); gotoxy (30,6);printf("LA SERIE ES:"); if (l==0 || l==1) { printf(" %.0f",1); s = recursum(1);printf(" \n"); getch(); Imagen ejecución Dev C++ ■ C:\Users\Home\Documents\1SEMESTRE\PERIODO 2022 A\PROGRAMACION 1\ALGORITMOS\DEBERES\DEBER 6\ejercicios en C\seriefi Verificación APLICACION GENERA SERIE DE NUMEROS DE FIBONACCI INGRESE EL LIMITE DE LA SERIE: 14 del

LA SERIE ES: 1 1 2 3 5 8 13 21 34 55 89 144 233 377

Process exited after 5.679 seconds with return value 0

Presione una tecla para continuar . .

algoritmo

	Desarrollo		Etapa
Determinar si un	Determinar si un número es primo.		
Entrada Num E	Proceso	Salida	
entero	Si n/2 <= j Entonces  p=1  Sino Si n mod j = 0 Entonces  p=0  Sino  p=nuprimo(n,j+1)  Fin si	p E entero	Análisis del problema
Si n/2 ≤ p=1 Sino Fin FinSi FinFuncion Algoritmo r Definir Escribi Leer num Escribi Si num=1 Escr SiNo s=nuprim Si s=1 E Escr SINO	sune Singaña 7-2022  uprimo(n,j)  p Como Entero  j Entonces  Si n mod j = 0 Entonces  p=0  Sino p=nuprimo(n,j+1) si  ecurprimo  s Como Real num Como Real r"APLICACION DETERMINA SI UN NUMERO ES P r"INGRESE EL NUMERO: " Sin Saltar  r "EL NUMERO ",num Sin Saltar o num=4 o num=0 Entonces sibir " NO ES PRIMO "  o (num,2) intonces sibir " ES PRIMO "	RIMO "	Diseño del algoritmo

```
Imagen ejecución pseint

▶ PSelnt - Ejecutando proceso RECURPRIMO

*** Ejecución Iniciada. ***

APLICACION DETERMINA SI UN NUMERO ES PRIMO
INGRESE EL NUMERO: > 5
EL NUMERO 5 ES PRIMO

*** Ejecución Finalizada. ***

APLICACION DETERMINA SI UN NUMERO ES PRIMO
INGRESE EL NUMERO: > 4
EL NUMERO 4 NO ES PRIMO

*** Ejecución Finalizada. ***

APLICACION DETERMINA SI UN NUMERO ES PRIMO
INGRESE EL NUMERO: > 4
EL NUMERO 4 NO ES PRIMO

*** Ejecución Finalizada. ***
```

```
Imagen obtenida en Dev c++
#include<stdio.h>
#include<conio2.h>
int nuprimo(int n,int j);
/* autor: Josune Singaña */
/* fecha:24-07-2022 */
int nuprimo(int n, int j) {
    int p;
if (n/2<=j) {
 p = 1;
     } else {
    if (n%j==0) {
        p = 0;
} else {
p = nuprimo(n,j+1);
    return p;
int main() {
         textbackground(7);
    clrscr();
     textcolor(3);
                                                                                                          Diseño del
     int num, s;
    char opc;
                                                                                                          algoritmo
        gotoxy (40,2);printf("APLICACION DETERMINA SI UN NUMERO ES PRIMO \n");
    opc ='s';
     while(opc=='S'||opc=='s')
         textcolor(8);
    printf("\n\tINGRESE EL NUMERO: ");
scanf("%1",&num);
         textcolor(LIGHTMAGENTA);
    printf("\n\tEL NUMERO: %i", num, " ");
if (num==1 || num==4|| num==0 ){
printf(" NO ES PRIMO \n");
     }else{
     s = nuprimo(num,2);
    if (s==1) {
    printf(" ES PRIMO \n");
     } else {
         printf(" NO ES PRIMO \n");
              textcolor(2);
         printf(" \n\t DESEA REPETIR EL PROCESO: S O N");
     opc=getch();
```

# Imagen ejecución Dev C++ □ C\Users\Home\Document\\Isemestre\Periodo 2022 A\Programacion I\algoritmos\Deberes\Deber 6\ejercicios en C\recurprimo.exe APLICACION DETERMINA SI UN NUMERO ES PRIMO INGRESE EL NUMERO: 23 EL NUMERO: 23 ES PRIMO DESEA REPETIR EL PROCESO: S O N INGRESE EL NUMERO: 6 EL NUMERO: 6 NO ES PRIMO DESEA REPETIR EL PROCESO: S O N INGRESE EL NUMERO: 4 EL NUMERO: 4 NO ES PRIMO DESEA REPETIR EL PROCESO: S O N Process exited after 19.39 seconds with return value 0 Presione una tecla para continuar . . .