

```
Program AROPERA;
{ VER 1 / JUN 4 2022
PROGRAM CREATED BY: JOSE GALINDO - WWW.AI20.ME - SHEWILLFIND.ME - GABRIELS.WORK -
MAY 2022
VERSION ADAPTADA DE: PROGRAM MEARTHE (*)
https://dontmissit665019001.files.wordpress.com/2022/05
/rational-computer-programmer-defends-himself-from
-irrational-people-s-slander.docx
Video of this program running:
Here https://youtu.be/CdDRD94BmAI
And a here (*put URL as a whole):
https://videos.files.wordpress.com/XYZyaFNZ/
aropera-program-that-creates-algebraic-formulas
-and-obtains-the-unknown-value.mp4
```

```
PARA PERMITIR GENERAR CADENAS CON SIMBOLOS PARA OPERACIONES ARITMETICAS BASICAS
}
{$M 8192, 8192,655360}
Uses Crt, DOS;
```

```
type
    regmem = ^ eledim;

    eledim = record
        pal: string[70];
        sig: regmem
    end;

Const
    maxwords: longint = 12000;

Var
    raropera1,raropera2:longint;
    inilista, eleact, eleant: regmem;
    laf,paginaciones, idl,inipag: longint;
    wtext,ratext: text;
    totalpag: longint;
    salirya:boolean;
    filai, cint1, cint2, x1, x2, gen : Longint;
    x3, x4,x5,x6,xx5,xx6, yy,zz,elcolort, ntiendme, gabrielswork, ai20me,
    shewillfindme,s4c,s5c,s6c : Longint;
    nrolista,lvxp, fxp,cuni, s7c,xx7,s8c,xx8,s9c,xx9: Longint;
    rep,repu : boolean;
    datos: array[1..42] of Longint;
    queteclafue,tipa, tecla, tecla3, teclan,teclal, confirma : Char;
    esvocal,ocentrado, encontrada,tpress: boolean;
    laletra2,laletra, subme: char;
    ubicador,ubicador2, genx, xcen, lomenu, lomenu2: Longint;
    vocales: array[1..5] of char;
    opcionesdelmenu: array[1..14] of string[41];
    encolores,inai,intento, iint,iint2:longint;
    palabrasl7: array[1..7] of string[12];
    consonantes: array[1..21] of char;
    arrdigitos: array[1..10] of char;
    cosu: array[1..200,1..2] of string[25];
    sesu: array[1..21] of string[25];
    resultadoaa,lista,lp,lpg,lpgper:string;
    llp:char;
    cs,vs: longint; {cs= controlador de secuencia, vs= valor secuencia, una idea para
    reducir codificacion del programa cuando combino silabas}
    xlp,tlp:longint;
    combinacion,
    modo3letras,m2l,m3l,m4l,proenglishm3l,proenglishm4l,proenglishm5l,m5l,m6l,m7l,m8l,m
    9l,cc,cv,i, serieunica:Longint;
    proenglishm6l,proenglishm7l: Longint;
    inicv,idiom:longint;
```

```

cantdigit:longint;
digitn:longint;
signo,cadedigit:string;
banderaesc:boolean;
numerodenumeros,numerodecolumnas:longint;
function verificaexistencia(archivo: String): boolean; forward;

function cadenapersonalizada(laforma:string;tamttotal:byte):string; forward;
Procedure cargaenpantalla(var nropantalla:longint); forward;

function verificaexistencia(archivo: String): boolean;
var
    a: text;
begin
    assign(a,archivo);
    {$I-}
    reset(a);
    {$I+}
    if (IOResult = 0) then
    begin
        close(a);
        verificaexistencia:=true;
    end
    else
        verificaexistencia:=false;
end;

Procedure Inicialistad;
Begin
idl:=0;
inilista:=nil;
End;

procedure anadirlistadin(palabra: string);
begin
idl:=idl+1;

if idl=1 then
begin
new(elect);
elect^.pal:=palabra;
elect^.sig:=Nil;
inilista:=elect;
elect:=Nil;
end
else
case idl of
2: begin
new(elect);
elect^.pal:=palabra;
elect^.sig:=Nil;
inilista^.sig:=elect;
elect:=elect;
end
else
begin
new(elect);
elect^.pal:=palabra;
elect^.sig:=Nil;
elect^.sig:=elect;
elect:=elect;
end;
end;{end of case idl}
if idl mod 23 = 0 then

```

```

    totalpag:=(idl div 23)
else
    totalpag:=(idl div 23)+1;
end;

Procedure mostrarlistad(var dde: longint);
var
sigp:longint;
ffe:char;

begin
clrscr;
cargaenpantalla(dde);
repeat

gotoxy(5,1);write('Content of text file :WNTXT.TXT');
gotoxy(1,25);clreol;write('Page ',dde,'/',totalpag,' ,Press PgUP/Home PgDw/End to
navigate or ESC to exit');
Repeat
    ffe:=readkey;
    if ffe=#0 then
        ffe:=readkey;
    Until ffe in[#27,#73,#81,#71,#79];
    case ffe of
        #73 : begin {pag up}
                if dde>1 then
                    dde:=dde-1;
                    cargaenpantalla(dde);
                end;
        #81 : begin {pag down}
                if dde<totalpag then
                    dde:=dde+1;
                    cargaenpantalla(dde);
                end;
        #71 : begin {Home / Inicio}
                dde:=1;
                cargaenpantalla(dde);;
            end;
        #79 : begin {End / Final}
                dde:=totalpag;
                cargaenpantalla(dde);
            end;
    end;
until (ffe=#27);
End;

Procedure cargaenpantalla(var nropantalla:longint);
var cdfi,defg:longint;
regmos:regmem;
compc,recorrido:longint;
Begin
if nropantalla=totalpag then
for compc:=2 to 24 do
begin
gotoxy(1,compc);clreol;
end;
compc:=0;

for recorrido:=1 to nropantalla do
BEGIN
cdfi:=0;
for defg:=1 to 23 do
begin

```

```

compc:=compc+1;
cdfi:=cdfi+1;
if compc=1 then
begin
  regmos:=inilista;
end
else
  case compc of
  2: BEGIN
    regmos:=inilista^.sig;
    END;

  else
    BEGIN
      regmos:=regmos^.sig;
      END;
  end;{del case y del if}
if recorrido=nropantalla then
begin
  gotoxy(5,1+cdfi);clreol;
  write(regmos^.pal)
end;
if cdfi=23 then
  cdfi:=0;
if compc=idl then
  break;
end;
if compc=idl then
  break;
end;
end;

```

```

Procedure leearchivoycrealistadinamica;
Var prec:string;
Begin
reset(ratext);
while not(eof(ratext)) do
begin
  readln(ratext,prec);
  anadirlistadin(prec);
end;
close(ratext);
End;
{Inicio}

```

```

Procedure paginarcontenidodearchivo;
var qh:byte;
    espr:char;
Begin
qh:=0;
Clrscr;
paginaciones:=0;
inipag:=1;
salirya:=false;
Inicialistad;
assign(ratext,'WNTXT.TXT');
if verificaexistencia('WNTXT.TXT') = False then
begin
  write('File WNTXT.TXT does not exist, create it with choice #1 in main Menu');
  espr:=readkey;
  qh:=1;
end

```

```

else
begin
  leearchivoycrealistadinamica;
  mostrarlistad(inipag);
end;
if qh=0 then
  Dispose(elect);
End;

function partedecimalaentero(reet:real;dii:byte):integer;
var
tfac,ho:integer;
tee:integer;
tre:real;
begin
tfac:=1;
tre:=reet-int(reet);
for ho:=1 to dii do
  tfac:=tfac*10;
tre:=tre*tfac;
tee:=trunc(tre);
partedecimalaentero:=tee;
end;

function cadenaanumero(digitos: string;tamano:longint):longint;
var
millones,cmiles,dmiles, miles, centenas, decenas, unidades: longint;
begin
millones:=0;
cmiles:=0;
dmiles:=0;
miles:=0;
centenas:=0;
decenas:=0;
unidades:=0;
case tamano of
7:begin
  millones:=ord(digitos[1])-48;
  cmiles:=ord(digitos[2])-48;
  dmiles:=ord(digitos[3])-48;
  miles:=ord(digitos[4])-48;
  centenas:=ord(digitos[5])-48;
  decenas:=ord(digitos[6])-48;
  unidades:=ord(digitos[7])-48;
end;
6:begin
  cmiles:=ord(digitos[1])-48;
  dmiles:=ord(digitos[2])-48;
  miles:=ord(digitos[3])-48;
  centenas:=ord(digitos[4])-48;
  decenas:=ord(digitos[5])-48;
  unidades:=ord(digitos[6])-48;
end;
5:begin
  dmiles:=ord(digitos[1])-48;
  miles:=ord(digitos[2])-48;
  centenas:=ord(digitos[3])-48;
  decenas:=ord(digitos[4])-48;
  unidades:=ord(digitos[5])-48;
end;
4:begin
  miles:=ord(digitos[1])-48;
  centenas:=ord(digitos[2])-48;
  decenas:=ord(digitos[3])-48;

```

```

    unidades:=ord(digitos[4])-48;
end;
3:begin
    centenas:=ord(digitos[1])-48;
    decenas:=ord(digitos[2])-48;
    unidades:=ord(digitos[3])-48;
end;
2:begin
    decenas:=ord(digitos[1])-48;
    unidades:=ord(digitos[2])-48;
end;
1: unidades:=ord(digitos[1])-48;
end;
cadenaanumero:=unidades+decenas*10+centenas*100+miles*1000+dmiles*10000+cmiles*1000
00+millones*1000000;
end;

```

```

function numeroacadena(numero: longint):string;

```

```

var
x1,x2:byte;
fd,cc,p1,lc,enb: longint;
rs:real;
cf:string;
begin

```

```

    cc:=1;
    fd:=0;
    cf:='';
    for x1:=1 to 6 do
    begin
        if numero>=fd then
            cc:=x1
        else break;
        if fd=0 then
            fd:=1;
        fd:=fd*10;
    end;

```

```

    fd:=1;
    for p1:=1 to cc-1 do
        fd:=fd*10;
    enb:=numero;
    repeat
        lc:=enb div fd;
        cf:=cf+chr(48+lc);
        enb:=enb mod fd;
        fd:=fd div 10;
    until (fd<1);
    numeroacadena:=cf+' ';
end;

```

```

function numeroacadena2(numero: longint):string;

```

```

var
x1,x2:byte;
fd,cc,p1,lc,enb: longint;
rs:real;
cf:string;
begin

```

```

    cc:=1;
    fd:=0;
    cf:='';
    for x1:=1 to 6 do
    begin
        if numero>=fd then

```

```

        cc:=x1
    else break;
    if fd=0 then
        fd:=1;
        fd:=fd*10;
end;
fd:=1;
for pl:=1 to cc-1 do
    fd:=fd*10;
enb:=numero;
repeat
lc:=enb div fd;
cf:=cf+chr(48+lc);
enb:=enb mod fd;
fd:=fd div 10;
until (fd<1);
numeroacadena2:=cf;
end;

```

```

procedure leedigitos(var digitos: string; var ttamano:longint);
var x:longint;
base:longint;
cequi,cequi2:string[8];
t:char;
i,s:longint;
salir:boolean;
begin
s:=0;
cequi:='';
repeat
repeat
t:=readkey;
until (t in['0'..'9']) or (t=#13) or (t=#8);
if (t in['0'..'9']) and (s<7) then
begin
s:=s+1;
cequi:=cequi+t;
write(t);
end
else
if (t=#8) and (s>0) then
begin
cequi2:='';
for x:=1 to s-1 do
begin
cequi2:=cequi2+cequi[x];
end;
cequi:=cequi2;
gotoxy(wherex-s,wherey);
clreol;write(cequi);
s:=s-1;
end
until (t=#13);
ttamano:=s;
digitos:=cequi;
end;

```

```

procedure limpiazona(ci,fi,cf,ff: longint); forward;

```

```

procedure menuhorizontal(fila,combisilabas: longint; var lacombinacion:longint;
colorfondo:byte);forward;

```

```

function detvocal(leva :char): boolean;

```

```

var ff:longint;
hallado: boolean;
begin
hallado:=false;
for ff:=1 to 5 do
  if vocales[ff]=leva then
    begin
      hallado:=true;
      detvocal:=true;
      break;
    end;
end;

if hallado=false then
  for ff:=1 to 22 do
    if consonantes[ff]=leva then
      begin
        hallado:=true;
        detvocal:=false;
        break;
      end;
end;
end;

function detm2(a :Longint): boolean;
begin
if a mod 2 = 0 then
  detm2:=true
else
  detm2:=false;
end;

procedure cuadrolineas(tipl: char; coi,fii,cof, fif: longint);
var
itc: longint;
horizontal,vertical, supizq,supder,infizq,infder:char;
begin
if tipl='s' then
begin
horizontal:=chr(196);vertical:=chr(179);
supizq:=chr(218);supder:=chr(191);infizq:=chr(212);infder:=chr(217);
end
else
begin
horizontal:=chr(205);vertical:=chr(186);
supizq:=chr(201);supder:=chr(187);infizq:=chr(200);infder:=chr(188);
end;
gotoxy(coi,fii);
write(supizq);
for itc:=1 to ((cof-coi)-1) do
  write(horizontal);
write(supder);
gotoxy(coi,fif);
write(infizq);
for itc:=1 to ((cof-coi)-1) do
  write(horizontal);
write(infder);
for itc:=1 to ((fif-fii)-1) do
begin
gotoxy(coi,fii+itc);write(vertical);
gotoxy(cof,fii+itc);write(vertical);
end;
end;

procedure generaetra(ev :boolean; var ll:char); forward;

```



```

procedure imprimesilaba(cgen, nds : Longint);
begin
if cgen=1 then
for x5:=1 to nds do
begin
esvocal:=detm2(x5);
genera letra(esvocal, la letra);
write(la letra);
end
else
for x5:=1 to nds do
begin
esvocal:=not(detm2(x5));
genera letra(esvocal, la letra);
write(la letra);
end;
end;

procedure imprimesilabayc(cgen, nds : Longint; var silaco: string; visu:longint);
var silacoaux: string;
begin
silacoaux:=silaco;
if cgen=1 then
for x5:=1 to nds do
begin
esvocal:=detm2(x5); {empieza en consonante, cuando x5 es 1, es impar, detm2 es
false,
y esvocal toma el valor de false, y al no ser vocal generara
una consonante}
genera letra(esvocal, la letra2);
if visu=1 then
write(la letra2);
silaco:=silaco+la letra2;
end
else
for x5:=1 to nds do
begin
esvocal:=not(detm2(x5)); {empieza en vocal, cuando x5 es 1, es impar, detm2 es
false, pero al negarlo,
lo hago true, y esvocal toma el valor de true, y al ser
vocal generara una vocal}
genera letra(esvocal, la letra2);
if visu=1 then
write(la letra2);
silaco:=silaco+la letra2;
end;
end;

procedure genera letra(ev :boolean; var ll:char);
var genl:longint;
begin
cv:=0;
cc:=0;
if ev=true then
begin
{genero una vocal al azar
ejem: ll:='A'}
genl:=Random(5)+1;
ll:=vocales[genl];
end
else
begin
{genero una consonante al azar
ejem: ll:='B'}

```

```

    genl:=Random(21)+1; {la posicion 22 era la ENIE ya no la considero}
    ll:=consonantes[genl];
end;
end;

function colorquemegusta(a :Longint): boolean;
{Descarto los colores que no me agradan, en este
caso los mas faciles para la vista considerando
fondo de pantalla negro, los codigos de colores en Turbo Pascal
son:
Black = 0, Blue = 1, Green = 2, Cyan = 3,
Red = 4, Magenta = 5, Brown = 6, LightGray = 7,
DarkGray = 8, LightBlue = 9, LightGreen = 10, LightCyan = 11,
LightRed = 12, LightMagenta= 13, Yellow = 14, White = 15
}
begin
case a of
  1,4,5,6,8,13: colorquemegusta:=false;
else
  colorquemegusta:=true;
end;
end;

procedure inicialettras;
var vst: string;
begin
cc:=0;
cv:=0;
{guardo en las matrices de vocales y consonantes mayusculas, las vocales y las
consonantes}
for i:=65 to 90 do
case i of
65, 69, 73, 79, 85: begin
                    cv:=cv+1;
                    vocales[cv]:=chr(i);
                    end;
else
  begin
    cc:=cc+1;
    consonantes[cc]:=chr(i);
  end;
end;
{consonantes[22]:=chr(165);
omitire la enie a causa de mi desconocimiento de como hacerlo en free pascal,
como consonantes considedare 21 ya no 22}
{aprovecho este procedimiento y inicio mi arreglo de digitos}
for i:=1 to 10 do
begin
  str(i-1,vst);
  arrdigitos[i]:=vst[1];
end;
end;

Procedure palabradecincoletras(m5l:BYTE; inicv:byte);
BEGIN
encolores:=1;

case m5l of
  1: begin {p(2)+p(2)+p(1)}

        for vs:=1 to 3 do
        begin

```

```

    case vs of
      1: cs:=2;
      2: cs:=2;
      3: cs:=1;
    end;

    if vs=1 then {esto hago para que la directiva solo se cumpla al inicio de la
palabra
    es decir en la 1ra silaba}
    case inicv of
      1:gen:=1;{empezara en consonante}
      2:gen:=2;{empezara en vocal}
      3:gen:=random(2)+1;{empezara en vocal o consonante segun el azar}
    end;

    if gen=1 then
      repeat
        elcolort:=random(7)+1;
      until colorquemegusta(elcolort)=true
    else
      repeat
        elcolort:=random(7)+9;
      until colorquemegusta(elcolort)=true;
    textcolor(elcolort);
    imprimesilabayc(gen,cs,lpg,encolores); {el numero de letras en la silaba sera
cs}

    end;{for del valor de secuencia (vs)}

end; {p(2)+p(2)+p(1)}

2: begin {p(2)+p(1)+p(2)}

    for vs:=1 to 3 do
    begin
      case vs of
        1: cs:=2;
        2: cs:=1;
        3: cs:=2;
      end;

      if vs=1 then {esto hago para que la directiva solo se cumpla al inicio de la
palabra
      es decir en la 1ra silaba}
      case inicv of
        1:gen:=1;{empezara en consonante}
        2:gen:=2;{empezara en vocal}
        3:gen:=random(2)+1;{empezara en vocal o consonante segun el azar}
      end;

      if gen=1 then
        repeat
          elcolort:=random(7)+1;
        until colorquemegusta(elcolort)=true
        else
          repeat
            elcolort:=random(7)+9;
          until colorquemegusta(elcolort)=true;
        textcolor(elcolort);
        imprimesilabayc(gen,cs,lpg,encolores); {el numero de letras en la silaba sera
cs}

        end;{for del valor de secuencia (vs)}

```

```

end; {p(2)+p(1)+p(2)}

3: begin {p(1)+p(2)+p(2)}

    for vs:=1 to 3 do
    begin
        case vs of
            1: cs:=1;
            2: cs:=2;
            3: cs:=2;
        end;

        if vs=1 then {esto hago para que la directiva solo se cumpla al inicio de la
palabra
es decir en la 1ra silaba}
        case inicv of
            1:gen:=1;{empezara en consonante}
            2:gen:=2;{empezara en vocal}
            3:gen:=random(2)+1;{empezara en vocal o consonante segun el azar}
        end;

        if gen=1 then
            repeat
                elcolort:=random(7)+1;
            until colorquemegusta(elcolort)=true
            else
                repeat
                    elcolort:=random(7)+9;
                until colorquemegusta(elcolort)=true;
            textcolor(elcolort);
            imprimesilabayc(gen,cs,lpg,encolores); {el numero de letras en la silaba sera
cs}

        end;{for del valor de secuencia (vs)}

    end; {p(1)+p(2)+p(2)}

4: begin {p(1)+p(1)+p(2)+p(1)}

    for vs:=1 to 4 do
    begin
        case vs of
            1: cs:=1;
            2: cs:=1;
            3: cs:=2;
            4: cs:=1;
        end;

        if vs=1 then {esto hago para que la directiva solo se cumpla al inicio de la
palabra
es decir en la 1ra silaba}
        case inicv of
            1:gen:=1;{empezara en consonante}
            2:gen:=2;{empezara en vocal}
            3:gen:=random(2)+1;{empezara en vocal o consonante segun el azar}
        end;

        if gen=1 then
            repeat
                elcolort:=random(7)+1;
            until colorquemegusta(elcolort)=true
            else
                repeat
                    elcolort:=random(7)+9;

```

```

    until colorquemegusta(elcolort)=true;
textcolor(elcolort);
imprimesilabayc(gen,cs,lpg,encolores); {el numero de letras en la silaba sera
cs}

    end;{for del valor de secuencia (vs)}

end; {p(1)+p(1)+p(2)+p(1)}

5: begin {p(1)+p(2)+p(1)+p(1)}

    for vs:=1 to 4 do
    begin
        case vs of
            1: cs:=1;
            2: cs:=2;
            3: cs:=1;
            4: cs:=1;
        end;

        if vs=1 then {esto hago para que la directiva solo se cumpla al inicio de la
palabra
es decir en la 1ra silaba}
        case inicv of
            1:gen:=1;{empezara en consonante}
            2:gen:=2;{empezara en vocal}
            3:gen:=random(2)+1;{empezara en vocal o consonante segun el azar}
        end;

        if gen=1 then
            repeat
                elcolort:=random(7)+1;
            until colorquemegusta(elcolort)=true
        else
            repeat
                elcolort:=random(7)+9;
            until colorquemegusta(elcolort)=true;
        textcolor(elcolort);
        imprimesilabayc(gen,cs,lpg,encolores); {el numero de letras en la silaba sera
cs}

        end;{for del valor de secuencia (vs)}

        end; {p(1)+p(2)+p(1)+p(1)}

6: begin {p(3)+p(2)}

    for vs:=1 to 2 do
    begin
        case vs of
            1: cs:=3;
            2: cs:=2;
        end;

        if vs=1 then {esto hago para que la directiva solo se cumpla al inicio de la
palabra
es decir en la 1ra silaba}
        case inicv of
            1:gen:=1;{empezara en consonante}
            2:gen:=2;{empezara en vocal}
            3:gen:=random(2)+1;{empezara en vocal o consonante segun el azar}
        end;

```

```

if gen=1 then
  repeat
    elcolort:=random(7)+1;
  until colorquemegusta(elcolort)=true
else
  repeat
    elcolort:=random(7)+9;
  until colorquemegusta(elcolort)=true;
textcolor(elcolort);
imprimesilabayc(gen,cs,lpg,encolores); {el numero de letras en la silaba sera
cs}

end;{for del valor de secuencia (vs)}

end; {p(3)+p(2)}

7: begin {p(2)+p(3)}

for vs:=1 to 2 do
begin
  case vs of
    1: cs:=2;
    2: cs:=3;
  end;

  if vs=1 then {esto hago para que la directiva solo se cumpla al inicio de la
palabra
  es decir en la 1ra silaba}
  case inicv of
    1:gen:=1;{empezara en consonante}
    2:gen:=2;{empezara en vocal}
    3:gen:=random(2)+1;{empezara en vocal o consonante segun el azar}
  end;

if gen=1 then
  repeat
    elcolort:=random(7)+1;
  until colorquemegusta(elcolort)=true
else
  repeat
    elcolort:=random(7)+9;
  until colorquemegusta(elcolort)=true;
textcolor(elcolort);
imprimesilabayc(gen,cs,lpg,encolores); {el numero de letras en la silaba sera
cs}

end;{for del valor de secuencia (vs)}

end; {p(2)+p(3)}

8: begin {p(3)+p(1)+p(1)}

for vs:=1 to 3 do
begin
  case vs of
    1: cs:=3;
    2: cs:=1;
    3: cs:=1;
  end;

  if vs=1 then {esto hago para que la directiva solo se cumpla al inicio de la
palabra
  es decir en la 1ra silaba}
  case inicv of

```

```

1:gen:=1;{empezara en consonante}
2:gen:=2;{empezara en vocal}
3:gen:=random(2)+1;{empezara en vocal o consonante segun el azar}
end;

if gen=1 then
  repeat
    elcolort:=random(7)+1;
  until colorquemegusta(elcolort)=true
else
  repeat
    elcolort:=random(7)+9;
  until colorquemegusta(elcolort)=true;
textcolor(elcolort);
imprimesilabayc(gen,cs,lpg,encolores); {el numero de letras en la silaba sera
cs}

end;{for del valor de secuencia (vs)}

end; {p(3)+p(1)+p(1)}

9: begin {p(1)+p(3)+p(1)}

  for vs:=1 to 3 do
  begin
    case vs of
      1: cs:=1;
      2: cs:=3;
      3: cs:=1;
    end;

    if vs=1 then {esto hago para que la directiva solo se cumpla al inicio de la
palabra
  es decir en la 1ra silaba}
    case inicv of
      1:gen:=1;{empezara en consonante}
      2:gen:=2;{empezara en vocal}
      3:gen:=random(2)+1;{empezara en vocal o consonante segun el azar}
    end;

    if gen=1 then
      repeat
        elcolort:=random(7)+1;
      until colorquemegusta(elcolort)=true
    else
      repeat
        elcolort:=random(7)+9;
      until colorquemegusta(elcolort)=true;
    textcolor(elcolort);
    imprimesilabayc(gen,cs,lpg,encolores); {el numero de letras en la silaba sera
cs}

    end;{for del valor de secuencia (vs)}

    end; {p(1)+p(3)+p(1)}

10: begin {p(1)+p(1)+p(3)}

  for vs:=1 to 3 do
  begin
    case vs of
      1: cs:=1;
      2: cs:=1;
      3: cs:=3;

```

```

    end;

    if vs=1 then {esto hago para que la directiva solo se cumpla al inicio de la
palabra
    es decir en la 1ra silaba}
    case inicv of
    1:gen:=1;{empezara en consonante}
    2:gen:=2;{empezara en vocal}
    3:gen:=random(2)+1;{empezara en vocal o consonante segun el azar}
    end;

    if gen=1 then
    repeat
    elcolort:=random(7)+1;
    until colorquemegusta(elcolort)=true
    else
    repeat
    elcolort:=random(7)+9;
    until colorquemegusta(elcolort)=true;
    textcolor(elcolort);
    imprimesilabayc(gen,cs,lpg,encolores); {el numero de letras en la silaba sera
cs}

    end;{for del valor de secuencia (vs)}

    end; {p(1)+p(1)+p(3)}

11: begin {p(4)+p(1)}

    for vs:=1 to 2 do
    begin
    case vs of
    1: cs:=4;
    2: cs:=1;
    end;

    if vs=1 then {esto hago para que la directiva solo se cumpla al inicio de la
palabra
    es decir en la 1ra silaba}
    case inicv of
    1:gen:=1;{empezara en consonante}
    2:gen:=2;{empezara en vocal}
    3:gen:=random(2)+1;{empezara en vocal o consonante segun el azar}
    end;

    if gen=1 then
    repeat
    elcolort:=random(7)+1;
    until colorquemegusta(elcolort)=true
    else
    repeat
    elcolort:=random(7)+9;
    until colorquemegusta(elcolort)=true;
    textcolor(elcolort);
    imprimesilabayc(gen,cs,lpg,encolores); {el numero de letras en la silaba sera
cs}

    end;{for del valor de secuencia (vs)}

    end; {p(4)+p(1)}

12: begin {p(1)+p(4)}

    for vs:=1 to 2 do

```



```

begin
  case vs of
    1: cs:=4;
    2: cs:=1;
  end;
  if vs=1 then {esto hago para que la directiva solo se cumpla al inicio de la
palabra
  es decir en la 1ra silaba}
  case inicv of
    1:gen:=1;{empezara en consonante}
    2:gen:=2;{empezara en vocal}
    3:gen:=random(2)+1;{empezara en vocal o consonante segun el azar}
  end;

  if gen=1 then
    repeat
      elcolort:=random(7)+1;
    until colorquemegusta(elcolort)=true
  else
    repeat
      elcolort:=random(7)+9;
    until colorquemegusta(elcolort)=true;
  textcolor(elcolort);
  imprimesilabayc(gen,cs,lpg,encolores); {el numero de letras en la silaba sera
cs}

  end;{for del valor de secuencia (vs)}

  end; {p(4)+p(1)}

```

```
13: begin {p(5)}
```

```

  case inicv of
    1:gen:=1;{empezara en consonante}
    2:gen:=2;{empezara en vocal}
    3:gen:=random(2)+1;{empezara en vocal o consonante segun el azar}
  end;

  if gen=1 then
    repeat
      elcolort:=random(7)+1;
    until colorquemegusta(elcolort)=true
  else
    repeat
      elcolort:=random(7)+9;
    until colorquemegusta(elcolort)=true;
  textcolor(elcolort);
  imprimesilabayc(gen,5,lpg,encolores);

  end; {p(5)}
end;{case MODOF}

```

```
END; {PROCEDURE palabras 5 letras}
```

```

procedure limpiaazona(ci,fi,cf,ff: longint);
var ca,fa,i,j:longint;
begin
ca:=wherex;
fa:=wherey;
for i:=ci to cf do
  for j:=fi to ff do
    begin

```

```

        gotoxy(i,j);
        if (i=80) and (j=25) then
            clreol
        else
            write(' ');
    end;
gotoxy(ca,fa);
end;

```

```

procedure menuhorizontal(fila,combisilabas: longint; var lacombinacion:longint;
colorfondo:byte);
{el menu que produce no siempre sera horizontal, segun el numero de elementos
tambien puede ser vertical}
var
orientacionhorizontal:boolean;
tmenu,recorrido: longint;
quetecla:char;
seeligioalgo:boolean;
begin
orientacionhorizontal:=true;
idiom:=2;{por defecto general}
inicv:=3;{por defecto al azar}
for recorrido:=1 to 13 do
    opcionesdelmenu[recorrido]:='';
seeligioalgo:=false;
case combisilabas of
1:
begin
    opcionesdelmenu[1]:='Classic (5 letters size words)';
    opcionesdelmenu[2]:='Customize words and size';
    tmenu:=2;
end;
2:
begin
    opcionesdelmenu[1]:='p2';
    opcionesdelmenu[2]:='p1p1';
    opcionesdelmenu[3]:='All';
    tmenu:=3;
end;
3:
begin
    opcionesdelmenu[1]:='p2p1';
    opcionesdelmenu[2]:='p1p2';
    opcionesdelmenu[3]:='p1p1p1';
    opcionesdelmenu[4]:='p3';
    opcionesdelmenu[5]:='All';
    tmenu:=5;
end;
4:
begin
    opcionesdelmenu[1]:='p2p2';
    opcionesdelmenu[2]:='p2p1p1';
    opcionesdelmenu[3]:='p1p2p1';
    opcionesdelmenu[4]:='p1p1p2';
    opcionesdelmenu[5]:='p4';
    opcionesdelmenu[6]:='All';
    tmenu:=6;
end;
5:
begin
    opcionesdelmenu[1]:='p2p2p1';
    opcionesdelmenu[2]:='p2p1p2';
    opcionesdelmenu[3]:='p1p2p2';
    opcionesdelmenu[4]:='p1p1p2p1';

```

```

opcionesdelmenu[5]:='p1p2p1p1';
opcionesdelmenu[6]:='p3p2';
opcionesdelmenu[7]:='p2p3';
opcionesdelmenu[8]:='p3p1p1';
opcionesdelmenu[9]:='p1p3p1';
opcionesdelmenu[10]:='p1p1p3';
opcionesdelmenu[11]:='p4p1';
opcionesdelmenu[12]:='p1p4';
opcionesdelmenu[13]:='p5';
opcionesdelmenu[14]:='All';
tmenu:=14;
orientacionhorizontal:=false;
end;
6:
begin
opcionesdelmenu[1]:='p1p2p2p1';
opcionesdelmenu[2]:='p1p2p1p2';
opcionesdelmenu[3]:='p2p1p1p2';
opcionesdelmenu[4]:='p2p1p2p1';
opcionesdelmenu[5]:='p1p1p2p2';
opcionesdelmenu[6]:='p2p2p1p1';
opcionesdelmenu[7]:='p1p1p2p1p1';
opcionesdelmenu[8]:='p2p2p2';
opcionesdelmenu[9]:='p6';
opcionesdelmenu[10]:='All';
tmenu:=10;
orientacionhorizontal:=false;
end;
7:
begin
opcionesdelmenu[1]:='p1p1p2p2p1';
opcionesdelmenu[2]:='p1p2p2p1p1';
opcionesdelmenu[3]:='p1p1p2p1p2';
opcionesdelmenu[4]:='p1p2p1p1p2';
opcionesdelmenu[5]:='p1p2p1p2p1';
opcionesdelmenu[6]:='p2p1p1p2p1';
opcionesdelmenu[7]:='p2p1p2p1p1';
opcionesdelmenu[8]:='p1p2p2p2';
opcionesdelmenu[9]:='p2p1p2p2';
opcionesdelmenu[10]:='p2p2p1p2';
opcionesdelmenu[11]:='p2p2p2p1';
opcionesdelmenu[12]:='p7';
opcionesdelmenu[13]:='All';
tmenu:=13;
orientacionhorizontal:=false;
end;
21:
begin
opcionesdelmenu[1]:='1 #s';
opcionesdelmenu[2]:='1..8#s (random)';
opcionesdelmenu[3]:='5 #s';
opcionesdelmenu[4]:='7 #s';
opcionesdelmenu[5]:='8 #s';
tmenu:=5;
end;
22:
begin
opcionesdelmenu[1]:='1 #s';
opcionesdelmenu[2]:='1..7#s (random)';
opcionesdelmenu[3]:='5 #s';
opcionesdelmenu[4]:='7 #s';
tmenu:=4;
end;
23:

```

```

begin
  opcionesdelmenu[1]:= '1 #s';
  opcionesdelmenu[2]:= '1..5#s (random)';
  opcionesdelmenu[3]:= '5 #s';
  tmenu:=3;
end;
24:
begin
  opcionesdelmenu[1]:= '1 #s';
  opcionesdelmenu[2]:= '1..4#s (random)';
  opcionesdelmenu[3]:= '4 #s';
  tmenu:=3;
end;
25:
begin
  opcionesdelmenu[1]:= '1 #s';
  opcionesdelmenu[2]:= '1..3#s (random)';
  opcionesdelmenu[3]:= '3 #s';
  tmenu:=3;
end;
26:
begin
  opcionesdelmenu[1]:= '1 #s';
  opcionesdelmenu[2]:= '1..2#s (random)';
  opcionesdelmenu[3]:= '2 #s';
  tmenu:=3;
end;
100:
begin
  opcionesdelmenu[1]:= 'Starts consonant';
  opcionesdelmenu[2]:= 'Starts Vowel';
  opcionesdelmenu[3]:= 'Any can start';
  tmenu:=3;
end;
101:
begin
  opcionesdelmenu[1]:= 'Focus English';
  opcionesdelmenu[2]:= 'General ';
  tmenu:=2;
end;
199:
begin
  opcionesdelmenu[1]:= 'Overwrite WNTXT.TXT';
  opcionesdelmenu[2]:= 'Append WNTXT.TXT';
  tmenu:=2;
end;
200:
begin
  opcionesdelmenu[1]:= 'Overwrite ratext.TXT';
  opcionesdelmenu[2]:= 'Append ratext.TXT';
  tmenu:=2;
end;
201:
begin
  opcionesdelmenu[1]:= '1000 (ms)';
  opcionesdelmenu[2]:= '500 (ms)';
  opcionesdelmenu[3]:= '250 (ms)';
  opcionesdelmenu[4]:= '0 (ms)';
  tmenu:=4;
end;
202:
begin
  opcionesdelmenu[1]:= '14 ';
  opcionesdelmenu[2]:= '25 ';

```

```

opcionesdelmenu[3]:= '70 ';
opcionesdelmenu[4]:= '230 ';
opcionesdelmenu[5]:= '1260 ';
opcionesdelmenu[6]:= 'Other ';
tmenu:=6;
end;
203:
begin
opcionesdelmenu[1]:= 'Series of random content ';
opcionesdelmenu[2]:= 'Load file WNTXT.TXT ';
opcionesdelmenu[3]:= 'Exit ';
tmenu:=3;
orientacionhorizontal:=false;
end;
end; {case combisilabas}
lacombinacion:=1;

Repeat
gotoxy(1, fila);
textbackground(black);
textcolor(white);
clreol;
if orientacionhorizontal then
write('Use arrows ', chr(27), '/', chr(26), ' & enter ')
else
write('Use arrows ', chr(24), '/', chr(25), ' & enter ');
for recorrido:=1 to tmenu do
begin
if recorrido=lacombinacion then
begin
textbackground(colorfondo);
textcolor(black)
end
else
begin
textbackground(black);
textcolor(lightgreen);
end;
if orientacionhorizontal=false then
gotoxy(40, fila+recorrido-1);

write(opcionesdelmenu[recorrido]);
if recorrido<tmenu then
begin
textcolor(colorfondo);
write('.');
end;
end;
repeat
quetecla:=readkey;
if quetecla=#13 then
seeligioalgo:=true;
if quetecla=#0 then
quetecla:=readkey;
{#72 arriba, #80 abajo, #75 izquierda, #77 derecha}
case orientacionhorizontal of
true:
case quetecla of
#75: if lacombinacion=1 then
lacombinacion:=tmenu
else
lacombinacion:=lacombinacion-1;
#77: if lacombinacion=tmenu then
lacombinacion:=1

```

```

        else
            lacombinacion:=lacombinacion+1;
        end;
false:{el menu debe ser vertical}
case quetecla of
    #72: if lacombinacion=1 then
        lacombinacion:=tmenu
    else
        lacombinacion:=lacombinacion-1;
    #80: if lacombinacion=tmenu then
        lacombinacion:=1
    else
        lacombinacion:=lacombinacion+1;
    end;
end; {de la evaluacion de la orientacionhorizontal}
until (quetecla in [#75,#77,#72,#80]) or (seeligioalgo=true);
until (seeligioalgo=true);
textbackground(black);
textcolor(cyan);
if orientacionhorizontal=false then
    limpiazona(40,6,52,20);
end;

```

```

Procedure ejecutaopc4;
{Esta basado en mi programa t7n}
Var
a4,a5,mododepalabras,crso,b,b0,b1,b2,y,t1,linea:longint;
fi,uvi,upo,video,post:longint;
quefue,tecla:char;
valorquefue:byte;
matrizuv: array[1..12] of longint;
filacondatorepetido:boolean;
vocaloconsonante,estructura:longint;
lineag:string[75];
lsupe,linfe:longint;
auxlaestru, laestru,ledlp:string[33]; {la estructura de la palabra}
tmledlp:byte; {tamano max de la estructura de la palabra}
posdelasterisco:byte; {esto me va a decir en que posicion esta el asterisco, si en
4 o en la 69 x ejm..}
posdelslash:byte; {esto me va a decir en que posicion esta el slash}
Begin
inicialetras;
clrscr;
assign(wtext,'WNTXT.TXT'); {archivo de texto ya tiene nombre}
gotoxy(1,24);clreol;write('The words in lists words will be saved at WNTXT.TXT');
if verificaexistencia('WNTXT.TXT') then {llamo a la funcion que verifica la
existencia del archivo}
    begin
        gotoxy(1,25);menuhorizontal(25,199,crso, CYAN);
        if crso=1 then
            rewrite(wtext) {lo creamos}
        else append(wtext); {los abrimos para agregar registros}
        end
    else
        rewrite(wtext); {lo creamos}
Repeat
clrscr;
gotoxy(10,10);write('Especify the range for the random #s');
gotoxy(10,11);write('From #:');readln(linfe);
gotoxy(10,12);write('To #:');readln(lsupe);
if lsupe<=linfe then

```

```

begin
  write('Input again because # are equal or the high# limit is minor than the low#
limit');
  tecla:=readkey;
end;
Until (lsupe>linfe);
clrscr;
{cambiosaqui}
menuhorizontal(2,1,mododepalabras,cyan);
case mododepalabras of
1: {modo clasico de 5 letras como en la version del 17/Mayo/2022}
begin
  clrscr;
  gotoxy(7,1);write('Choose quantity of #s from (' ,linfe,'-',lsupe,') per row:');
  menuhorizontal(2,21,numeroenumeros,cyan);
  case numeroenumeros of
{ se considera: opcionesdelmenu[1]:='1 #s';
  opcionesdelmenu[2]:='1..8#s (random)';
  opcionesdelmenu[3]:='5 #s'; opcionesdelmenu[4]:='7 #s';
  opcionesdelmenu[5]:='8 #s'; tmenu:=5;}
1:numerodecolumnas:=1;
2:numerodecolumnas:=random(8)+1;
3:numerodecolumnas:=5;
4:numerodecolumnas:=7;
5:numerodecolumnas:=8;
end;
  gotoxy(7,3);write('Choose the structure for the word of 5 letters. ');
  menuhorizontal(4,5,estructura, brown);
  limpiazona(40,5,52,5);
  if estructura=14 then
    estructura:=random(13)+1;
  gotoxy(40,4);clreol;
  gotoxy(40,5);clreol;
  gotoxy(40,4);write(opcionesdelmenu[estructura]);
  auxlaestru:=opcionesdelmenu[estructura];
  menuhorizontal(23,100,vocaloconsonante, brown);
  auxlaestru:=auxlaestru+', '+opcionesdelmenu[vocaloconsonante];
end;
2:{nuevo modo, el personalizado}
begin
  Gotoxy(23,3); write('Input the structure for the random word. ');
  gotoxy(5,5); write('This are the elements to consider in the word structure:');
  Gotoxy(5,6); write(' (ALT+35)# =Any digit from 0 to 9, (ALT+38)&=Uppercase vowel');
  Gotoxy(5,7); write(' (ALT+36)$ =Uppercase consonant,
(ALT+39)',CHR(ORD(39)),'=Lowercase vowel');
  Gotoxy(5,8); write(' (ALT+95)_ =Lowercase consonant, (ALT+32)" "=One Blank
space');
  Gotoxy(5,9); write(' (ALT+124)| =Any Uppercase letter, (ALT+64)@=Any Lowercase
letter');
  Gotoxy(5,10);write(' (ALT+126)~ =Any consonant upper/lowercase');
  Gotoxy(5,11);write(' (ALT+60)< =Any Vowel upper/lowercase');
  Gotoxy(5,12);write(' (ALT+63)?=Any (#, &, $, ',CHR(ORD(39)),'_','" ',|,@,~,<)'');
  Gotoxy(5,13);write(' (ALT+34)"=Any except blank
space(#, &, $, ',CHR(ORD(39)),'_',|,@,~,<)'');
  Gotoxy(1,14);write(' (ALT+96)`=As ? till the end: Any character from that point
till the end');
  Gotoxy(1,15);write(' (ALT+92)\=As " till the end: Any character but blank space
from that point till the end');
  Gotoxy(7,16);write('Note: if you include (`) or (\) you will be asked to specify
word size');
  Gotoxy(7,17);write('if none ` or /, the size will the size of the word (Max 33
characters)');
  Gotoxy(7,18);write('Example: A word structure:&$ ##',CHR(ORD(39)),' may produce a
word such as:ER 70a');

```

```

Gotoxy(1,19);write('|+##=##,@+##=##,|*#=##,*#=##,|*##=###,*##=### will save
results for | or @');
repeat
gotoxy(3,20);clreol;write('Build word structure and press Enter:');readln(ledlp);
until (ledlp<>'') and (length(ledlp)<=33);
posdelslash:=pos('\',ledlp);
posdelasterisco:=pos('\',ledlp);
if (posdelslash<>0) or (posdelasterisco<>0) then
begin
if (posdelslash<>0) and (posdelasterisco<>0) then
begin
if posdelasterisco>posdelslash then
begin
quefue:='\';
valorquefue:=posdelslash;
end
else
begin
quefue:='\';
valorquefue:=posdelasterisco;
end;
end
else
begin
if (posdelslash<>0) and (posdelasterisco=0) then
begin
quefue:='\';
valorquefue:=posdelslash;
end;
if (posdelslash=0) and (posdelasterisco<>0) then
begin
quefue:='\';
valorquefue:=posdelasterisco;
end;
end;
end;
repeat
gotoxy(3,21);clreol;write('As you put a ',quefue,' in position ',valorquefue);
write(' specify the total length of the word (Max 33):');
readln(tmledlp);
if (tmledlp<valorquefue) or (tmledlp>33) then
writeln('As ',quefue,' is in position ',valorquefue,' the word length must be
between ',valorquefue,'-33');
until (tmledlp>=valorquefue) and (tmledlp<=33);
end
else
begin
tmledlp:=length(ledlp);
if tmledlp>33 then
tmledlp:=33; {fijo el tamaño max de la palabra a 33, mi número elegido (para
joder a los hipócritas)}
end;
lpgper:=cadenapersonalizada(ledlp,tmledlp);
writeln('Sample of random word with that structure (press any key to continue:');
{cambiosaquí}
write(lpgper);
tecla:=readkey;
clrscr;
gotoxy(7,1);write('Choose quantity of #s from (' ,linfe,'-',lsupe,') per row:');
case tmledlp of
1..7: begin
menuhorizontal(2,21,numerodenumeros,cyan);
case numerodenumeros of
{ se considera: opcionesdelmenu[1]:='1 #s';
opcionesdelmenu[2]:='1..8#s (random)';

```



```

    opcionesdelmenu[3]:='5 #s'; opcionesdelmenu[4]:='7 #s';
    opcionesdelmenu[5]:='8 #s'; tmenu:=5;}
    1:numerodecolumnas:=1;
    2:numerodecolumnas:=random(8)+1;
    3:numerodecolumnas:=5;
    4:numerodecolumnas:=7;
    5:numerodecolumnas:=8;
    end;
end;
8..13: begin
    menuhorizontal(2,22,numerodenumeros,cyan);
    case numerodenumeros of
    { se considera: opcionesdelmenu[1]:='1 #s';
    opcionesdelmenu[2]:='1..7#s (random)';
    opcionesdelmenu[3]:='5 #s';
    opcionesdelmenu[4]:='7 #s'; tmenu:=4;}
    1:numerodecolumnas:=1;
    2:numerodecolumnas:=random(6)+1;
    3:numerodecolumnas:=5;
    4:numerodecolumnas:=7;
    end;
end;
14..20: begin
    menuhorizontal(2,23,numerodenumeros,cyan);
    case numerodenumeros of
    { se considera: opcionesdelmenu[1]:='1 #s';
    opcionesdelmenu[2]:='1..5#s (random)';
    opcionesdelmenu[3]:='5 #s';
    tmenu:=3;}
    1:numerodecolumnas:=1;
    2:numerodecolumnas:=random(5)+1;
    3:numerodecolumnas:=5;
    end;
end;
21..25: begin
    menuhorizontal(2,24,numerodenumeros,cyan);
    case numerodenumeros of
    { se considera: opcionesdelmenu[1]:='1 #s';
    opcionesdelmenu[2]:='1..4#s (random)';
    opcionesdelmenu[3]:='4 #s';
    tmenu:=3;}
    1:numerodecolumnas:=1;
    2:numerodecolumnas:=random(4)+1;
    3:numerodecolumnas:=4;
    end;
end;
26..30: begin
    menuhorizontal(2,25,numerodenumeros,cyan);
    case numerodenumeros of
    { se considera: opcionesdelmenu[1]:='1 #s';
    opcionesdelmenu[2]:='1..3#s (random)';
    opcionesdelmenu[3]:='3 #s';
    tmenu:=3;}
    1:numerodecolumnas:=1;
    2:numerodecolumnas:=random(3)+1;
    3:numerodecolumnas:=3;
    end;
end;
31..33: begin
    menuhorizontal(2,26,numerodenumeros,cyan);
    case numerodenumeros of
    { se considera: opcionesdelmenu[1]:='1 #s';
    opcionesdelmenu[2]:='1..2#s (random)';
    opcionesdelmenu[3]:='2 #s';

```

```

        tmenu:=3;}
        1:numerodecolumnas:=1;
        2:numerodecolumnas:=random(2)+1;
        3:numerodecolumnas:=2;
    end;
end;
end;
end; {del case mododepalabras}
linea:=0;
b0:=1;
clrscr;
gotoxy(15,7);write('*** The Lines of random numbers from ',linfe,' to ',lsupe,'
*****');
gotoxy(15,10);write('The 1st page will start in #:');readln(fi);
repeat
randomize;
if b0 mod 21 = 0 then
b1:=b0 div 21
else
b1:=(b0 div 21) + 1;
clrscr;
textbackground(black);textcolor(lightgreen);
if mododepalabras=1 then
    laestru:=auxlaestru
else
    laestru:=ledlp;
Gotoxy(1,1);write('Programmed by Jose Galindo, author of: AI20.ME, SHEWILLFIND.ME,
GABRIELS.WORK');
Gotoxy(1,2);write('c22cd@hotmail.com +51 930208791- Word Structure:',laestru);
textbackground(blue);
textcolor(black);
Gotoxy(1,3);write('Pag',b1:6,' Random word + random #s ',linfe,'-',lsupe,' each /
ESC=exit, other for next');
textcolor(white);textbackground(black);
for y:=1 to 21 do
begin
    gotoxy(1,3+y);
    lpg:='';
    lpgper:='';
    resultadoaa:='';
    signo:='';
    if mododepalabras=1 then {se hace como antes, la version previa}
        palabradecincoletras(estructura,vocaloconsonante) {este procedimiento actualiza
la variable global lpg }
    else {caso contrario es 2 y se considera la palabra personalizada segun la nueva
version, pues se eligio el modo 2}
        begin
            lpgper:=cadenapersonalizada(ledlp,tmledlp);
            {Esto es algo novedoso}
            if (laestru='|+##=##') or (laestru='@+##=##') then
                begin
                    raroperal:=cadenaanumero(copy(lpgper,3,2),2);
                    raropera2:=cadenaanumero(copy(lpgper,6,2),2);
                    if raroperal>raropera2 then
                        begin
                            signo:='-';
                            resultadoaa:=' '+copy(lpgper,1,1)+'='+signo+numeroacadena(raroperal-
raropera2);
                        end
                    else
                        begin
                            signo:='';

```

```

                resultadoaa:='
'+copy(lpgper,1,1)+'=' +signo+numeroacadena (raropera2-raroperal);
                end;
        end
    else
        if (laestru='|*#=##') or (laestru='@*#=##') then
            begin
                raroperal:=cadenaanumero(copy(lpgper,3,1),1);
                raropera2:=cadenaanumero(copy(lpgper,5,2),2);
                if raroperal<>0 then
                    begin
                        if (raropera2 mod raroperal) >0 then {hay decimales}
                            begin

signo:='.'+numeroacadena(partedecimalaentero(raropera2/raroperal,2));
                resultadoaa:=' '+copy(lpgper,1,1)+'=' +numeroacadena2 (raropera2
div raroperal)+signo;
                    end
                    else {no hay decimales}
                        resultadoaa:=' '+copy(lpgper,1,1)+'=' +numeroacadena (raropera2
div raroperal);
                    end
                    else {se evito division entre cero}
                        resultadoaa:=' '+copy(lpgper,1,1)+'=div/0';
                    end
                end
            else
                if (laestru='|*##=###') or (laestru='@*##=###') then
                    begin
                        raroperal:=cadenaanumero(copy(lpgper,3,2),2);
                        raropera2:=cadenaanumero(copy(lpgper,6,3),3);
                        if raroperal<>0 then
                            begin
                                if (raropera2 mod raroperal) >0 then {hay decimales}
                                    begin

signo:='.'+numeroacadena(partedecimalaentero(raropera2/raroperal,2));
                resultadoaa:=' '+copy(lpgper,1,1)+'=' +numeroacadena2 (raropera2
div raroperal)+signo;
                    end
                    else {no hay decimales}
                        resultadoaa:=' '+copy(lpgper,1,1)+'=' +numeroacadena (raropera2
div raroperal);
                    end
                    else {se evito division entre cero}
                        resultadoaa:=' '+copy(lpgper,1,1)+'=div/0';
                    end
                end;
            write(lpgper:tmlledlp);
        end;
        a4:=5;
        a5:=length(lpgper);
        if a5>a4 then
            a4:=a5;
        lineag:='';
        laf:=fi+21*(linea);
        lineag:=lineag+numeroacadena(laf)+' ';
        if mododepalabras=1 then
            lineag:=lineag+lpg
        else
            begin
                {lo novedoso sigue aqui}
                lineag:=lineag+lpgper;
                lineag:=lineag+resultadoaa;
            end;
        if y mod 2 = 0 then

```

```

    textcolor(lightblue)
else
    textcolor(lightgreen);
gotoxy(a4+2,3+y);write(laf,'#s ');
for t1:=1 to numerodecolumnas do
begin
    video:=random(lsupe-linfe+1)+linfe;
    {video:=random(1502)+101;}
    matrizuv[t1]:=video;
    gotoxy(a4+14+(t1-1)*6,3+y);write(video:4,' ');
    lineag:=lineag+' '+numeroacadena(video);
end;

b0:=b0+1;
filacondatorepetido:=false;
for b:=1 to numerodecolumnas do
    for b2:=numerodecolumnas downto 1 do
        if b<>b2 then
            begin
                if matrizuv[b2]=matrizuv[b] then
                    begin
                        filacondatorepetido:=true;
                        break;
                    end;
                if filacondatorepetido then break;
            end;
        if filacondatorepetido then
            begin
                write('repeated #s');
                fi:=fi-1;
                lineag:=lineag+' R';
            end;
        writeln(wtext, lineag); {grabamos fila / linea en archivo de texto}
        fi:=fi+1;
    end;
tecla:=readkey;
if tecla=#0 then tecla:=readkey;
until tecla=#27;
close(wtext);
End;

procedure generadigito(var eldigito:char);
var t:byte;
begin
    t:=random(10)+48;
    eldigito:=chr(t);
end;

function cadenapersonalizada(laforma:string;tamttotal:byte):string;
var posas,poss1,nv,ta,bx1,bx2:byte;
palabranueva: string[33];
caractergenerado:char;
valorespc: array[1..10] of char;
hefc, hefc2:boolean;
begin
    hefc:=false; {hasta el fin cualquier caracter}
    hefc2:=false;
    valorespc[1]:='#'; {9}
    valorespc[2]:='$'; {C}
    valorespc[3]:='&'; {V}
    valorespc[4]:=' '; { }
    valorespc[5]:=CHR(ORD(39)); {v}
    valorespc[6]:='_'; {c}
    valorespc[7]:='|'; {V/C}

```

```

valorespc[8]:='@'; {v/c}
valorespc[9]:='~'; {C/c}
valorespc[10]:='<'; {V/v}
palabranueva:='';
inicialetras;
posas:=pos('`',laforma);
possl:=pos('\',laforma);
if (posas=0) and (possl=0) then {no hay ni asterisco ni slash}
    ta:=length(laforma)
else
    ta:=tamtotal;
{entre ` y \ se considera al que se encuentre primero , mas a la izquierda en la
cadena}
for bx1:=1 to ta do
begin
    if (laforma[bx1]='`') then
        if hefc2=false then
            hefc:=true;
    if (laforma[bx1]='\') then
        if hefc=false then
            hefc2:=true;

    if (laforma[bx1]='?') or (hefc=true) then
begin
    nv:=random(10)+1;
    laforma[bx1]:=valorespc[nv];
end;
    if (laforma[bx1]='"') or (hefc2=true) then
begin
    repeat
        nv:=random(10)+1;
    until (nv<>4);
    laforma[bx1]:=valorespc[nv];
end;

case laforma[bx1] of
'#':generadigito(caractergenerado); {9}
'&':genera letra(true,caractergenerado); {V}
'$':genera letra(false,caractergenerado); {C}
' ':caractergenerado:=' ';
CHR(ORD(39)):begin {v}
    genera letra(true,caractergenerado);
    caractergenerado:=chr(ord(caractergenerado)+32);
end;
'_':begin {c}
    genera letra(false,caractergenerado);
    caractergenerado:=chr(ord(caractergenerado)+32);
end;
'|':begin {V/C}
    bx2:=random(2)+1;
    case bx2 of
    1:genera letra(true,caractergenerado);
    2:genera letra(false,caractergenerado);
    end;
end;
'@':begin {v/c}
    bx2:=random(2)+1;
    case bx2 of
    1:genera letra(true,caractergenerado);
    2:genera letra(false,caractergenerado);
    end;
    caractergenerado:=chr(ord(caractergenerado)+32);
end;
'~':begin {C/c}

```

```

    genera letra (false, caracter generado);
    bx2:=random(2)+1;
    if bx2=2 then
        caracter generado:=chr(ord(caracter generado)+32);
    end;
'<':begin {V/v}
    genera letra (true, caracter generado);
    bx2:=random(2)+1;
    if bx2=2 then
        caracter generado:=chr(ord(caracter generado)+32);
    end;
else caracter generado:=laforma[bx1];
end;
palabra nueva:=palabra nueva+caracter generado;
end;
cadena personalizada:=palabra nueva;
end;

```

```

{INICIO DEL PROGRAMA
PASCAL TIENE LA LIMITACION DE QUE EL PROGRAMA PRINCIPAL
NO PUEDE SER MUY LARGO EN CODIFICACION, A SABER HASTA 24K,
POR ESO HE ESTABLECIDO RUTINAS PARA CADA OPCION DEL MENU
PRINCIPAL
}

```

```

Begin
Randomize;
Repeat
    o centrado:=false;
    clrscr;
    gotoxy(3,7);write('AROPERA, by Jose Galindo (WWW.AI20.ME)-Lima-Peru, Jun 4th
2022');
    menu horizontal(10,203,lomenu, BLUE);
    case lomenu of
        1: ejecuta opc4;{lista de filas con una palabras y varios numeros aleatorios
c/u}
        2: paginar contenido de archivo;
    end;
until (lomenu=3);
End.

```