Codificação de Programas em Português Estruturado

```
Programa ALG1
Início
  Escreva "Bom dia"
Fim
```

```
Programa ALG2
Var
  X : Inteiro
Início
  Leia X
  Escreva X
Fim
```

```
Programa ALG3
Var
  X : Inteiro
  Y : Inteiro
Início
  Leia X
  Y ← X ↑ 2
  Escreva Y
Fim
```

```
Programa ALG4
Var
  X : Inteiro
  Y : Inteiro
  Z : Inteiro
Início
  Leia X
  Leia Y
  Z ← X + Y
  Escreva Z
Fim
```

```
Programa ALG5
Var
  X : Inteiro
  Y : Inteiro
  Z : Inteiro
Início
  Leia X
  Leia Y
  Z ← X ↑ 2 + Y ↑ 2
  Escreva Z
Fim
```

```
Programa ALG6
Var
  X : Inteiro
Início
  Leia X
  Se (X > 100) Então
    Escreva X
  Fim_se
Fim
```

```
Programa ALG7
Var
  X : Inteiro
  Y : Inteiro
  Z : Inteiro
Início
  Leia X
  Leia Y
  Se (X > 100) Então
    Z ← X + Y
    Escreva Z
  Fim_se
Fim
```

```
Programa ALG8
Var
  X : Inteiro
  Y : Inteiro
Início
  Leia X
  Leia Y
  Se (X <= Y) Então
    Escreva X
  Senão
    Escreva Y
  Fim_se
Fim
```

```
Programa ALG9
Var
  X : Inteiro
  Y : Inteiro
Início
  Leia X
  Se (X >= 10) Então
    Y ← X ↑ 2
  Senão
    Y ← X ↑ 3
  Fim_se
  Escreva Y
Fim
```

```
Programa ALG10
Var
  X  : Inteiro
  Y  : Inteiro
  N1 : Inteiro
  N2 : Inteiro
Início
  Leia X
  Leia Y
  Se (X > Y) Então
    N1 ← Y
    N2 ← X
  Senão
    N1 ← X
    N2 ← Y
  Fim_se
  Escreva N1
  Escreva N2
Fim
```

```
Programa ALG11
Var
  X : Inteiro
  I : Inteiro
Início
  X ← 0
  I ← 1
  Enquanto (I <= 10) Faça
    Escreva X
    X ← X + 2
    I ← I + 1
  Fim_enquanto
Fim
```

```
Programa ALG12
Var
  X : Inteiro
  I : Inteiro
Início
  X ← 1
  I ← 1
  Enquanto (I <= 10) Faça
    Escreva X
    X ← X * 2
    I ← I + 1
  Fim_enquanto
Fim
```

Programas em Linguagem PASCAL

```
Program ALG01;
Begin
  WriteLn('Bom dia');
End.
```

```
Program ALG02;
Var
  X : Integer;
Begin
  ReadLn(X);
  WriteLn(X);
End.
```

```
Program ALG03;
Var
  X : Integer;
  Y : Integer;
Begin
  ReadLn(X);
  Y := X * X;
  WriteLn(Y);
End.
```

```
Program ALG04;
Var
  X : Integer;
  Y : Integer;
  Z : Integer;
Begin
  ReadLn(X);
  ReadLn(Y);
  Z := X + Y;
  WriteLn(Z);
End.
```

```
Program ALG05;
Var
  X : Integer;
  Y : Integer;
  Z : Integer;
Begin
  ReadLn(X);
  ReadLn(Y);
  Z := X * X + Y * Y;
  WriteLn(Z);
End.
```

```
Program ALG06;
Var
  X : Integer;
Begin
  ReadLn(X);
  If (X > 100) Then
    WriteLn(X);
End.
```

```
Program ALG07;
Var
  X : Integer;
  Y : Integer;
  Z : Integer;
Begin
  ReadLn(X);
  ReadLn(Y);
  If (X > 100) Then
    Begin
      Z := X + Y;
      WriteLn(Z);
    End;
End.
```

```
Program ALG08;
Var
  X : Integer;
  Y : Integer;
Begin
  ReadLn(X);
  ReadLn(Y);
  If (X <= Y) Then
    WriteLn(X)
  Else
    WriteLn(Y);
End.
```

```
Program ALG09;
Var
  X : Integer;
  Y : Integer;
Begin
  ReadLn(X);
  If (X >= 10) Then
    Y := X * X
  Else
    Y := X * X * X;
  WriteLn(Y);
End.
```

```
Program ALG10;
Var
  X  : Integer;
  Y  : Integer;
  N1 : Integer;
  N2 : Integer;
Begin
  ReadLn(X);
  ReadLn(Y);
  If (X > Y) Then
    Begin
      N1 := Y;
      N2 := X;
    End
  Else
    Begin
      N1 := X;
      N2 := Y;
    End;
  WriteLn(N1);
  WriteLn(N2);
End.
```

```
Program ALG11;
Var
  X : Integer;
  I : Integer;
Begin
  X := 0;
  I := 1;
  While (I <= 10) Do
    Begin
      WriteLn(X);
      X := X + 2;
      I := I + 1;
    End;
End.
```

```
Program ALG12;
Var
  X : Integer;
  I : Integer;
Begin
  X := 1;
  I := 1;
  While (I <= 10) Do
    Begin
      WriteLn(X);
      X := X * 2;
      I := I + 1;
    End;
End.
```

Codificação de Programas em Linguagem Structured BASIC

```
REM ALG01
PRINT "Bom dia"
END
```

```
REM ALG02
DIM x AS INTEGER
INPUT x
PRINT x
END
```

```
REM ALG03
DIM x AS INTEGER
DIM y AS INTEGER
INPUT x
y = x ^ 2
PRINT y
END
```

```
REM ALG04
DIM x AS INTEGER
DIM y AS INTEGER
DIM z AS INTEGER
INPUT x
INPUT y
z = x + y
PRINT z
END
```

```
REM ALG05
DIM x AS INTEGER
DIM y AS INTEGER
DIM z AS INTEGER
INPUT x
INPUT y
z = x ^ 2 + y ^ 2
PRINT z
END
```

```
REM ALG06
DIM x AS INTEGER
INPUT x
IF (x > 100) THEN
    PRINT x
END IF
END
```

```
REM ALG07
DIM x AS INTEGER
DIM y AS INTEGER
DIM z AS INTEGER
INPUT x
INPUT y
IF (x > 100) THEN
    z = x + y
    PRINT z
END IF
END
```

```
REM ALG08
DIM x AS INTEGER
DIM y AS INTEGER
INPUT x
INPUT y
IF (x <= y) THEN
    PRINT x
ELSE
    PRINT y
END IF
END
```

```
REM ALG09
DIM x AS INTEGER
DIM y AS INTEGER
INPUT x
IF (x >= 10) THEN
    y = x ^ 2
ELSE
    y = x ^ 3
END IF
PRINT y
END
```

```
REM ALG10
DIM x AS INTEGER
DIM y AS INTEGER
DIM n1 AS INTEGER
DIM n2 AS INTEGER
INPUT x
INPUT y
IF (x > y) THEN
    n1 = y
    n2 = x
ELSE
    n1 = x
    n2 = y
END IF
PRINT n1
PRINT n2
END
```

```
REM ALG11
DIM x AS INTEGER
DIM i AS INTEGER
x = 0
i = 1
WHILE (i <= 10)
    PRINT x
    x = x + 2
    i = i + 1
WEND
END
```

```
REM ALG12
DIM x AS INTEGER
DIM i AS INTEGER
x = 1
i = 1
WHILE (i <= 10)
    PRINT x
    x = x * 2
    i = i + 1
WEND
END
```

Codificação de Programas em Linguagem C

```c
/* ALG01 */
#include <stdio.h>
int main()
{
  printf("Bom dia\n");
  return 0;
}
```

```c
/* ALG02 */
#include <stdio.h>
int X;
int main()
{
  scanf("%i", &X);
  printf("%i\n", X);
  return 0;
}
```

```c
/* ALG03 */
#include <stdio.h>
#include <math.h>
int X;
int Y;
int main()
{
  scanf("%i", &X);
  Y = pow(X, 2);
  printf("%i\n", Y);
  return 0;
}
```

```c
/* ALG04 */
#include <stdio.h>
int X;
int Y;
int Z;
int main()
{
  scanf("%i", &X);
  scanf("%i", &Y);
  Z = X + Y;
  printf("%i\n", Z);
  return 0;
}
```

```c
/* ALG05 */
#include <stdio.h>
#include <math.h>
int X;
int Y;
int Z;
int main()
{
  scanf("%i", &X);
  scanf("%i", &Y);
  Z = pow(X, 2) + pow(Y, 2);
  printf("%i\n", Z);
  return 0;
}
```

```c
/* ALG06 */
#include <stdio.h>
int X;
int main()
{
  scanf("%i", &X);
  if (X > 100)
    printf("%i\n", X);
  return 0;
}
```

```c
/* ALG07 */
#include <stdio.h>
int X;
int Y;
int Z;
int main()
{
  scanf("%i", &X);
  scanf("%i", &Y);
  if (X > 100)
    {
      Z = X + Y;
      printf("%i\n", Z);
    }
  return 0;
}
```

```c
/* ALG08 */
#include <stdio.h>
int X;
int Y;
int main()
{
  scanf("%i", &X);
  scanf("%i", &Y);
  if (X <= Y)
    printf("%i\n", X);
  else
    printf("%i\n", Y);
  return 0;
}
```

```c
/* ALG09 */
#include <stdio.h>
#include <math.h>
int X;
int Y;
int main()
{
  scanf("%i", &X);
  if (X >= 10)
    Y = pow(X, 2);
  else
    Y = pow(X, 3);
  printf("%i\n", Y);
  return 0;
}
```

```c
/* ALG10 */
#include <stdio.h>
int X;
int Y;
int N1;
int N2;
int main()
{
  scanf("%i", &X);
  scanf("%i", &Y);
  if (X > Y)
    {
      N1 = Y;
      N2 = X;
    }
  else
    {
      N1 = X;
      N2 = Y;
    }
  printf("%i\n", N1);
  printf("%i\n", N2);
  return 0;
}
```

```c
/* ALG11 */
#include <stdio.h>
int X;
int I;
int main()
{
  X = 0;
  I = 1;
  while (I <= 10)
    {
      printf("%i\n", X);
      X = X + 2;
      I = I + 1;
    }
  return 0;
}
```

```c
/* ALG12 */
#include <stdio.h>
int X;
int I;
int main()
{
  X = 1;
  I = 1;
  while (I <= 10)
    {
      printf("%i\n", X);
      X = X * 2;
      I = I + 1;
    }
  return 0;
}
```

Codificação em Linguagem Lua

```
-- ALG01
io.write("Bom dia\n")
```

```
-- ALG02
X = io.read("*number")
io.write(X, "\n")
```

```
-- ALG03
X = io.read("*number")
Y = X ^ 2
io.write(Y, "\n")
```

```
-- ALG04
X = io.read("*number")
Y = io.read("*number")
Z = X + Y
io.write(Z, "\n")
```

```
-- ALG05
X = io.read("*number")
Y = io.read("*number")
Z = X ^ 2 + Y ^ 2
io.write(Z, "\n")
```

```
-- ALG06
X = io.read("*number")
if (X > 100) then
  io.write(X, "\n")
end
```

```
-- ALG07
X = io.read("*number")
Y = io.read("*number")
if (X > 100) then
  Z = X + Y
  io.write(Z, "\n")
end
```

```
-- ALG08
X = io.read("*number")
Y = io.read("*number")
if (X <= Y) then
  io.write(X, "\n")
else
  io.write(Y, "\n")
end
```

```
-- ALG09
X = io.read("*number")
if (X >= 10) then
  Y = X ^ 2
else
  Y = X ^ 3
end
io.write(Y)
```

```
-- ALG10
X = io.read("*number")
Y = io.read("*number")
if (X > Y) then
  N1 = Y
  N2 = X
else
  N1 = X
  N2 = Y
end
io.write(N1, "\n")
io.write(N2, "\n")
```

```
-- ALG11
X = 0
I = 1
while (I <= 10) do
  io.write(X, "\n")
  X = X + 2
  I = I + 1
end
```

```
-- ALG12
X = 1
I = 1
while (I <= 10) do
  io.write(X, "\n")
  X = X * 2
  I = I + 1
end
```

Codificação em Linguagem Python

```
# ALG01
print("Bom dia")
```

```
# ALG02
X = int(input())
print(X)
```

```
# ALG03
X = int(input())
Y = X ** 2
print(Y)
```

```
# ALG04
X = int(input())
Y = int(input())
Z = X + Y
print(Z)
```

```
# ALG05
X = int(input())
Y = int(input())
Z = X ** 2 + Y ** 2
print(Z)
```

```
# ALG06
X = int(input())
if (X > 100):
  print(X)
```

```
# ALG07
X = int(input())
Y = int(input())
if (X > 100):
  Z = X + Y
  print(Z)
```

```
# ALG08
X = int(input())
Y = int(input())
if (X <= Y):
  print(X)
else:
  print(Y)
```

```
# ALG09
X = int(input())
if (X >= 10):
  Y = X ** 2
else:
  Y = X ** 3
print(Y)
```

```
# ALG10
X = int(input())
Y = int(input())
if (X > Y):
  N1 = Y
  N2 = X
else:
  N1 = X
  N2 = Y
print(N1)
print(N2)
```

```
# ALG11
X = 0
I = 1
while (I <= 10):
  print(X)
  X = X + 2
  I = I + 1
```

```
# ALG12
X = 1
I = 1
while (I <= 10):
  print(X)
  X = X * 2
  I = I + 1
```