**Algorithme du programme principal**

1. Début combinaison
2. Répéter

ecrire("Saisir un entier n="),lire(N)

ecrire("Saisir un entier p="),lire(P)

 Jusqu’à (P>=0)et(N>=P)

#### 2/ C🡸 FN Fact(N) div ( FN Fact(P) \*FN Fact (N-P))

3) Ecrire ("La combinaison est ",C)

4/ Fin combinaison

**Analyse du programme principal :**

**Nom** combinaison

**Résultat** = Ecrire ("La combinaison est ",C)

C🡸 FN Fact(N) div ( FN Fact (P) \*FN Fact (N-P))

N,P =[] Répéter

N= donnée("Saisir un entier n= ")

P= donnée("Saisir un entier p= ")

Jusqu’à (P>0)et(N>=P)

**Fin** combinaison

**Tableau de déclaration des objets globaux**

|  |  |  |
| --- | --- | --- |
| **Objet** | **Nature / Type** | **Rôle** |
| N, P,CFact | Var/ EntierFonction |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

**Analyse de la fonction Fact:**

**DEF FN Fact(a :entier) :entier**

**Resultat**= Fact🡨f

f=[ ] pour i de 2 à a faire

 f🡨f\*i

 fin pour

fin Fact

**Tableau de déclaration des objets locaux**

|  |  |  |
| --- | --- | --- |
| **Objet** | **Nature / Type** | **rôle** |
| i,f | Var/Entier |  |

**Algorithme de la fonction Fact**

0/ DEF FN Fact (a : entier) : entier

1/f=[f🡸1] Pour i de 1 à x faire

f🡨 f\*i

 Fin pour

2/ Fact 🡨 f

3/ Fin Fact

**Traduction en Turbo Pascal**

**PROGRAM combinaison;**

**USES WINCRT;**

**VAR N,P, C: integer;**

{\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* la fonction fact\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*}

**FUNCTION Fact (a: integer): integer;**

**VAR i :integer ;**

**BEGIN**

 **F:=1;**

 **For i:=2 to a do**

 **Begin**

 **f:= f\*i;**

 **End;**

 **Fact:= f;**

**End;**

{\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* le programme principal\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*}

**BEGIN**

**Repeat**

 **write ('Saisir un entier n='); readln(n);**

 **write ('Saisir un entier p='); readln(p);**

**Until (p>=0)and(n>p) ;**

**N) div (Fact(P)\*Fact(N-P));**

**Writeln ('La combinaison est ',C) ;**

**END.**