

Exercice 1 :

```
Algorithme exo1_1 ;
Var
    N, M, S, I, J : entier ;
    Trouve : booléen;
Debut
Lire (N);
M ← N +1;
S ← 2;
I ← 1;
Tant que (S < M) faire
    S ← S * 2;
    I ← I +1;
Fin Tant que;
Si (S=M) alors
    J ←2;
    Trouve ← Vrai ;
    Tant que (J <= (I div 2)) et (Trouve) faire
        Si (I div J = 0) alors
            Trouve ← Faux ;
        Sinon
            J ← J + 1 ;
        Fin Si ;
    Fin Tant que ;
Si (Trouve) alors
    Ecrire (n, ' est un nombre de Mersenne' ) ;
Sinon
    Ecrire (n, 'n'est pas un nombre de Mersenne' ) ;
Fin Si ;
Fin.
```

Algorithme exo1_2 ;

```
Var
    N, M, S, I, J : entier ;
    Trouve : booléen;
Debut
Pour N allant de 1 à 99 faire
    M ← N +1;
    S ← 2;
    I ← 1;
    Tant que (S < M) faire
        S ← S * 2;
        I ← I +1;
    Fin Tant que;
Si (S=M) alors
    J ←2;
    Trouve ← Vrai ;
    Tant que (J <= (I div 2)) et (Trouve) faire
        Si (I div J = 0) alors
```

```
        Trouve ← Faux ;
    Sinon
        J ← J + 1 ;
    Fin Si ;
    Fin Tant que ;
Si (Trouve) alors
    Ecrire (n, 'est un nombre de Mersenne' ) ;
Fin Si ;
Fin.
```

Exercice 2 :

Algorithme Exo2 ;

```
Var
    I, M, N, Z, X, Y : entier ;
    Trouve : Booléen ;
Début
    X ←11 ;
    Y ←13 ;
    Tant que (y <= 99) faire
        I ←2 ;
        Trouve ← Vrai ;
        Tant que (i < Y div 2) et (Trouve) faire
            Si (X mod I = 0) ou (Y mod i = 0) alors
                Trouve = Faux ;
            Fin Si ;
            I ← I+1 ;
        Fin Tant que ;
    Si (Trouve) alors
        Ecrire (X , 'et', Y , ' sont jumeaux' ) ;
    Fin Si ;
    X ← X + 1 ;
    Y ← Y + 1 ;
    Fin Tant que ;
Fin.
```