

Correction

Partie A :

1.

```
Fonction suivant(n: type entier)      # type du résultat : entier
Début
    Si n== 1  Alors
        Retourner(1)
    Sinon
        Si n%2== 0
            Retourner(n / 2)
        Sinon
            Retourner(3 * n + 1)
    FinSi
FinSi
FinFonction
```

2. Pour le nombre 14, il permet de récupérer l'ensemble des nombres de la suite de Syracuse dans une liste qui se terminera par 1.

3.

```
Fonction trajet(n: type entier)      # type du résultat : entier
Début
    liste= []  # liste vide
    TantQue suivant(n) != 1 Faire
        liste ← liste + [suivant(n)]
        n ← suivant(n)
    FinTantQue
    liste = liste + [1]
    Retourner liste
FinFonction
```

4.

```
Fonction MAX(liste: type liste)      # type du résultat : entier
Début
    M = liste[0]  # initialisation
    Pour i variant de 1 à len(liste)
        Si M < liste[i] Faire
            M = liste[i]
        FinSi
    FinPour
    Retourner M
FinFonction
```

Ou mieux pour la suite:

```
Fonction MAX(n : type entier)      # type du résultat : entier
Début
    M = 1  # initialisation
    TantQue suivant(n) != 1 Faire
        Si M < suivant(n) Faire
            M = suivant(n)
            n = suivant(n)+\bnvc
        FinSi
        n = suivant(n)
    FinTantQue
    Retourner M
FinFonction
```

5.

```
TantQue len(trajet(n)) !=20 Faire
    n←n+1
FinTantQue
```

Afficher "le nombre le plus petit est"+ n +" et l'altitude max est "+ MAX(n)

6.

...