

```
//
Ce fameux problème né à Syracuse aux Etats-Unis, a suscité une conjecture
qui n'est toujours pas démontrée.
Prenez un entier naturel quelconque. S'il est pair, divisez-le par 2.
S'il est impair, multipliez-le par 3 et ajoutez 1.
Continuez de la même façon...et arrêtez-vous quand vous arrivez à 1.
La conjecture suppose que l'on arrive toujours à 1.
On peut associer à chaque entier une hauteur,
qui est le nombre d'opérations nécessaires avant d'arriver à 1 exemple:
```

```
h(1)=0
h(32)=5 : 32 16 8 4 2 1
h(5)=5 : 5 16 8 4 2 1
h(9)=19 : 9 28 14 7 22 11 34 17 52 26 13 40 20 10 5 16 8 4 2 1
```

```
//
```

```
POUR Syracuse :n
DONNE "h 0 DONNE "k :n
TANTQUE (:k > 1) [
TESTE (ENTIER (:k / 2)) = (:k / 2)
SIVRAI DONNE "k :k / 2
SIFAUZ DONNE "k (3* :k)+1
DONNE "h :h + 1
]
REPONDRE :h
FIN
```

```
POUR parité :n
DONNE "k :n
DONNE "pair 0 DONNE "impair 0
TANTQUE (:k > 1) [
TESTE (ENTIER (:k / 2)) = (:k / 2)
SIVRAI [ DONNE "k :k / 2 DONNE "pair :pair + 1 ]
SIFAUZ [ DONNE "k (3* :k)+1 DONNE "impair :impair + 1 ]
]
REPONDRE :pair - :impair
FIN
```

```
POUR impair :n
DONNE "k :n
DONNE "impair 0
TANTQUE (:k > 1) [
TESTE (ENTIER (:k / 2)) = (:k / 2)
SIVRAI [ DONNE "k :k / 2 DONNE "impair :impair - 1 ]
SIFAUZ [ DONNE "k (3* :k)+1 DONNE "impair :impair + 1 ]
]
REPONDRE :impair
FIN
```

```
POUR graph :c :n
EFF EFFTXT CT FCC NOIRE
DONNE "ancienlieu [ -240 (-250) ]
SEGMENT [ -240 (-250) ] [ 250 (-250)]
SEGMENT [ -240 (-250) ] [ -240 (-50)]
DESSINETEXTE [ -265 (-70) 0] "S(n):
DESSINETEXTE [ -265 (-263) 0] :c
DESSINETEXTE [ -250 + :n (-263) 0] :c + :n
DONNE "sup 0 FCC BLEUE DONNE "inf 1E8
DONNE "xsup :c DONNE "xinf :c
DONNE "possup [ :c (:c + :n) 0 ] DONNE "posinf [ :c (:c + :n) 0 ]
REPETE :n [
DONNE "y Syracuse ( :c + BOUCLE )
SI :y >= :sup [
DONNE "possup PH PH -250 + BOUCLE (-250) + :y 0
DONNE "xsup BOUCLE
DONNE "sup :y
]
SI :y <= :inf [
DONNE "posinf PH PH -250 + BOUCLE (-250) + :y 0
]
```

```

                DONNE "xinf BOUCLE
                DONNE "inf :y
            ]
        SEGMENT :ancienlieu [ -240 + BOUCLE (-250) + :y ]
        DONNE "ancienlieu PH -240 + BOUCLE (-250) + :y
    ]
    FCC BLEUE
    DESSINETEXTE :possup PH PH PH [Ssup( ) :xsup ( ) =] :sup
    DESSINETEXTE :posinf PH PH PH [Sinf( ) :xinf ( ) =] :inf
    ECRISLIGNE PH PH PH [ valeur minimale : ] :inf "amplitude: :sup - :inf
    FIN

    POUR variation :c :n
    DONNE "sup 0 DONNE "mini 1E4
    REPETE :n [
        DONNE "y Syracuse (:c + BOUCLE )
        SI :y >= :sup DONNE "sup :y
        SI :y <= :mini DONNE "mini :y
    ]
    REpondRE ENTIER ((:sup - :mini) / :n )
    FIN

    POUR graph2 :c :n
    //EFF EFFTXT// CT FCC NOIRE
    DESSINETEXTE [ -51 56 0 ] [ Variations de Syracuse: ]
    DONNE "ancienlieu [ -240 (-20) ]
    SEGMENT [ -240 (-20) ] [ 250 (-20)]
    SEGMENT [ -240 (-20) ] [ -240 73]
    DESSINETEXTE [ -265 73 0] "Delta(n):
    DESSINETEXTE [ -265 (-25) 0] :c
    DESSINETEXTE [ -250 + :n (-25) 0] :c + :n
    DONNE "sup 0 DONNE "inf 1E5
    DONNE "xsup :c DONNE "xinf :c
    DONNE "possup [ :c (:c + :n) 0 ] DONNE "posinf [ :c (:c + :n) 0 ]
    FCC ROUGE
    REPETE :n [
        DONNE "y variation ( :c + BOUCLE ) 10
        SEGMENT :ancienlieu [ -240 + BOUCLE (-20) + :y ]
        DONNE "ancienlieu PH -240 + BOUCLE (-20) + :y
        SI :y >= :sup [
            DONNE "possup PH PH -255 + BOUCLE (20) + :y 0
            DONNE "xsup BOUCLE
            DONNE "sup :y
        ]
        SI :y <= :inf [
            DONNE "posinf PH PH -255 + BOUCLE (20) + :y 0
            DONNE "xinf BOUCLE
            DONNE "inf :y
        ]
    ]
    DESSINETEXTE :possup PH PH PH [Dsup( ) :xsup ( ) =] :sup
    DESSINETEXTE :posinf PH PH PH [Dinf( ) :xinf ( ) =] :inf
    FIN

    POUR graph3 :c :n
    //EFF EFFTXT// CT FCC NOIRE
    DESSINETEXTE [ -51 256 0 ] [ Etude de la parité: ]
    DONNE "ancienlieu [ -240 191]
    SEGMENT [ -240 191 ] [ 250 191]
    SEGMENT [ -240 147 ] [ -240 241]
    DESSINETEXTE [ 13 235 0] "Parité(n):
    DESSINETEXTE [ -265 185 0] :c
    DESSINETEXTE [ -250 + :n 185 0] :c + :n
    DONNE "sup 0 DONNE "inf 1E5
    DONNE "xsup :c DONNE "xinf :c
    DONNE "possup [ :c (:c + :n) 0 ] DONNE "posinf [ :c (:c + :n) 0 ]
    FCC MARRON
    REPETE :n [
        DONNE "y parité ( :c + BOUCLE )

```

```

SEGMENT :ancienlieu [ -240 + BOUCLE 191+ :y ]
DONNE "ancienlieu PH -240 + BOUCLE 191+ :y
SI :y >= :sup [
    DONNE "possup PH PH -255 + BOUCLE 191+ :y 0
    DONNE "xsup BOUCLE
    DONNE "sup :y
]
SI :y <= :inf [
    DONNE "posinf PH PH -255 + BOUCLE 191+ :y 0
    DONNE "xinf BOUCLE
    DONNE "inf :y
]
]
DESSINETEXTE :possup PH PH PH [Psup( ) :xsup( ) =] :sup
DESSINETEXTE :posinf PH PH PH [Pinf( ) :xinf( ) =] :inf

```

```

DONNE "ancienlieu [ -240 191]
DESSINETEXTE [ 13 145 0] "Impair(n):
DONNE "sup -1E5 DONNE "inf 0
DONNE "xsup :c DONNE "xinf :c
DONNE "possup [ :c (:c + :n) 0 ] DONNE "posinf [ :c (:c + :n) 0 ]
FCC ROSE
REPETE :n [

```

```

    DONNE "y impair ( :c + BOUCLE )
    SEGMENT :ancienlieu [ -240 + BOUCLE 191 + :y ]
    DONNE "ancienlieu PH -240 + BOUCLE 191 + :y
    SI :y >= :sup [
        DONNE "possup PH PH -255 + BOUCLE 191 - :y 0
        DONNE "xsup BOUCLE
        DONNE "sup :y
    ]
    SI :y <= :inf [
        DONNE "posinf PH PH -255 + BOUCLE 191+ :y 0
        DONNE "xinf BOUCLE
        DONNE "inf :y
    ]
]
DESSINETEXTE :possup PH PH PH [ Impsup( ) :xsup( ) =] :sup
DESSINETEXTE :posinf PH PH PH [ Impinf( ) :xinf( ) =] :inf
FIN

```

```

POUR tracer
graph 10 500 graph2 10 500 graph3 10 500
FIN

```

```

tracer // pour la plus récente version Gfx //

```

```

// Taper dans la ligne de commande (fenêtre de travail):

```

```

tracer

```

```

//

```