Fibonacci Nim

Function that returns the largest Fibonacci number less than or equal to a number *a*:

Define fib(a)=

Func

Local list

list:={1,1}

Loop

 If list[2]>a: Exit

 list:={list[2],list[2]+list[1]}

EndLoop

Return list[1]

EndFunc

Function that returns the Fibonacci representation of an integer *a*:

Define factorize(a)=

Func

Local n,list

list:={}

While a>0

 n:=fib(a)

 list:=augment({n},list)

 a:=a-n

EndWhile

Return list

EndFunc

Main program:

Define start()=

Prgm

Local n,mx,move,f,a,nspire

n:=0

While n<3

 Request "Starting number",n,0

 If n=0: Stop

 If n<3: Text "Minimum 3",0

EndWhile

nspire:=false

a:=0

While a<1 or a>2

 Request "1=Nspire begins, 2=You begin",a,0

 If a=0: Stop

EndWhile

If a=1: nspire:=true

mx:=n-1

Loop

 If nspire Then

 f:=factorize(n)

 move:=f[1]

 If move>mx: move:=1

 Text "There are "&string(n)&". Nspire takes "&string(move),0

 Else

 move:=0

 While move<1 or move>mx

 Request"There are "&string(n)&". You take? (max "&string(mx)&")",move,0

 If move=0: Stop

 EndWhile

 EndIf

 n:=n-move

 If n≤0: Exit

 nspire:=not nspire

 mx:=2\*move

EndLoop

If nspire Then

 Text "Nspire won",0

Else

 Text "You won",0

EndIf

EndPrgm