

Illustrate machine epsilon...

```
Eps=0.5  
while 1 != 1 + Eps  
    Eps /= 2  
    printbitsf64(1+Eps)  
end
```

The image displays a grid of 32 rows of binary strings. Each row consists of 60 bits. The first 11 bits of each row are '1's, followed by a '0', and then 48 '0's. The '1' and '0' bits are highlighted in green and blue respectively. The 48 trailing '0' bits are highlighted in red. The grid illustrates the machine epsilon of a floating-point system.

Stacktrace:

[1] top-level scope

@ In[6]:2

[2] eval

@ ./boot.jl:368 [inlined]

[3] include_string(mapexpr::typeof(REPL.softscope), mod::Module, code::String, filename::String)

@ Base ./loading.jl:1428

The @warn macro writes to the stderr channel:

```
println( $\pi^2$ )
@warn "Last warning!"
1 + 41
```

9.869604401089358

```
[ Warning: Last warning!
  @ Main In[7]:2
```

42