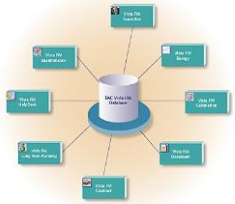
Aphorisms in Computer Science

Where you stand is where you sit

Requirements Engineering

- Requirements Engineering is more difficult now, because all systems that were easy to specify, have been built some time ago.
  - Tom DeMarco '01
- Requirements deficiencies are the primary source of project failures.
  - Robert Glass
- There are no wrong programs. Such a program is simply a different program
  - W. L. van der Poel

Modularization



- Every module is characterized by its knowledge of a design decision which it hides from all others. Its interface is chosen to reveal as little as possible about its inner workings.

- Only what is hidden can be changed without risk.

  - David L. Parnas '72

Dijkstra-isms

- The *goto* statement as it stands is just too primitive; it is too much an invitation to make a mess of one's program
  - E. W. Dijkstra '68

- Testing can show the presence, but not the absence of errors
  - E.W. Dijkstra

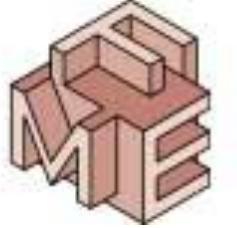
On validation and static verification

- We can conclude from experience that inspections increase productivity and improve final program quality.
  - M. E. Fagan '76

Errors

- Users don't observe errors or faults.
  They observe execution failures.
  - H. Mills '90
- Smaller changes have a higher error density than larger ones.
  - Basili-Möller
- Error prevention is better than error removal.
  - Mays

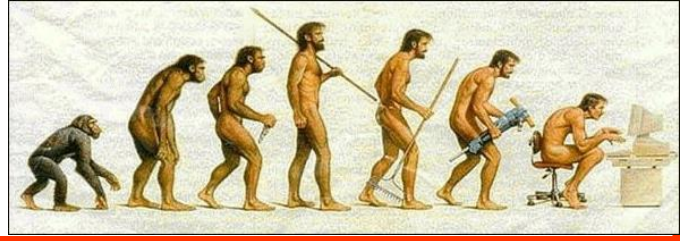
Formal Methods



- Formal methods significantly reduce design errors, or eliminate them early.
  - Bauer-Zemanek

- Proving programs solves the problems of correctness, documentation and compatibility.
  - C. A. R. Hoare

Software Evolution - 1



- The term *evolution* describes a process of progressive change in the attributes of entities. This may include:
  - improvement in some sense,
  - adaptation to a changing environment,
  - loss of not-required or undesired properties , or,
  - the emergence of new ones.

- M.M. Lehman '94

Software Evolution - 2

- A system that is used will be changed.
- An evolving system increases its complexity, unless work is done to reduce it.
- System evolution is determined by a feedback process.
  - M. M. Lehman

Complexity - 1

- The software field is not a simple one and, if anything, it is getting more complex at a faster rate than we can put in order.
  - Barry W. Boehm '79

- Building software will always be hard. There is inherently no silver bullet.
  - F.P. Brooks, Jr '87

Complexity - 2

- Programmers are always surrounded by complexity; we cannot avoid it. Our applications are complex because we are ambitious to use our computers in ever more sophisticated ways.
  - C.A.R. Hoare '81

- Simple, elegant solutions are more effective, but they are much harder to find than complex ones
  - N. Wirth '85

Complexity - 3

- If you have a procedure with 10 parameters, you probably missed some.
  - Alan Perlis

- The software is done.
  We are just trying to get it to work.
  - Statement in Executive Program Review

- Good, fast, cheap. Pick any two.
  - Old software engineering aphorism

Work power

- The best engineers or scientists do not work for a company, a university or a laboratory; they really work for themselves.
  - W. S. Humphrey '97

Software Architecture

- Software architecture involves the description of elements from which systems are built, interactions amongst those elements, patterns that guide there composition, and constraints on these patterns
  - Mary Shaw '96

- Architecture wins over technology
  - Morris-Ferguson

Performance

- The price/performance ratio of processors is halved every 18 months.
  - Moore
- The capacity of magnetic devices increases by a factor of ten every decade.
  - Hoagland
- Wireless bandwidth doubles every 2.5 years.
  - Cooper


Hello World

Hello World

from

the GNU archives

Hello world - 1

High School/Jr.High

```
10 PRINT "HELLO WORLD"
20 END
```

First year in College

```
program Hello(input, output)
begin
  writeln('Hello World')
end.
```

Senior year in College

```
(defun hello
  (print
    (cons 'Hello (list 'World))))
```

Hello World - 2

New professional

```
#include <stdio.h>
void main(void) {
  char *message[] = {"Hello ", "World"};
  int i;
  for(i = 0; i < 2; ++i)
    printf("%s", message[i]);
  printf("\n");
}
```

Hello World - 3a

Seasoned professional

```
#include <iostream.h>
#include <string.h>
class string
{
 private:
  int size;
  char *ptr;
 public:
  string() : size(0), ptr(new char('\0')) { }
```

Hello World - 3b

Seasoned professional (continued)

```
string(const string &s) : size(s.size)
 {
   ptr = new char[size + 1];
   strcpy(ptr, s.ptr);
 }
 ~string( ) { delete [] ptr; }

 friend
  ostream &operator <<(ostream &, const string &);
  string &operator=(const char *);
};
```

Hello World - 3c

Seasoned professional - continued(2)

```
 ostream &operator<<(ostream &stream, const string &s)
{   return(stream << s.ptr); }
  string &string::operator=(const char *chrs)
{ if (this != &chrs)
  { delete [] ptr;
    size = strlen(chrs);
    ptr = new char[size + 1];
    strcpy(ptr, chrs);
  }
  return(*this);
}
```

Hello World - 3d

Seasoned professional - continued(3)

```
string &string::operator=(const char *chrs)
{ if (this != &chrs)
  { delete [] ptr;
    size = strlen(chrs);
    ptr = new char[size + 1];
    strcpy(ptr, chrs);
  }
  return(*this);
}
```

Hello World - 3e

Seasoned professional - continued(4)

```
int main()
{
  string str;
  str = "Hello World";
  cout << str << endl;
  return(0);
}
```

Hello World - 4

System Administrator

```
#include <stdio.h>
#include <stdlib.h>
main() {
  char *tmp; int i=0; /* on y va bourin */
  tmp=(char *)malloc(1024*sizeof(char));
  while (tmp[i]="Hello Wolrd"[i++]);
  /* Ooopps y'a une infusion ! */
  i=(int)tmp[8];
  tmp[8]=tmp[9];
  tmp[9]=(char)i;
  printf("%s\n",tmp);
}
```

Hello World - 5

Apprentice Hacker

```
#!/usr/local/bin/perl
$msg="Hello, world.\n";
if ($#ARGV >= 0) {
  while(defined($arg=shift(@ARGV))) {
    $outfilename = $arg;
    open(FILE, ">" . $outfilename) || die "Can't write $arg:
$!\n";
    print (FILE $msg);
    close(FILE) || die "Can't close $arg: $!\n";
    }
  } else {  print ($msg); }
1;
```

Hello World - 6

Experienced Hacker

```
#include <stdio.h>
#include <string.h>
 #define S "Hello, World\n" main()
                          {exit(printf(S) == strlen(S) ? 0 : 1);}
```

Seasoned Hacker

```
% cc -o a.out ~/src/misc/hw/hw.c % a.out Hello, world.
```

Guru Hacker

```
% cat Hello, world.
```

Hello World - 7

New Manager (do you remember?)

```
10 PRINT "HELLO WORLD"
 20 END
```

Middle Manager

```
mail -s "Hello, world." bob@b12 Bob, could you please write
me a program that prints "Hello, world."? I need it by tomorrow.
^D
```

Senior Manager

```
% zmail jim I need a "Hello, world." program by this afternoon.
```

Hello World - 8

Chief Executive

% letter
letter: Command not found.
% mail To: ^X ^F ^C
% help mail help: Command not found.
% damn!
!: Event unrecognized
% logout

Hello World - 9

Research Scientist

```
PROGRAM HELLO
PRINT *, 'Hello World'
END
```

Older research Scientist

```
      WRITE (6, 100)
100 FORMAT (1H ,11HHELLO WORLD)
      CALL EXIT
      END
```

Hello World in RMI-context - 1

```
package examples.hello;
import java.rmi.Naming;
import java.rmi.RemoteException;
import java.rmi.RMISecurityManager;
import java.rmi.server.UnicastRemoteObject;
public class HelloImpl extends UnicastRemoteObject
implements Hello {
  public HelloImpl() throws RemoteException {
    super();
  }
  public String sayHello() {
    return "Hello World!";
  }
```

Hello World in RMI-context - 2

```
public static void main(String args[]) {
  // Create and install a security manager
  if (System.getSecurityManager() == null) {
    System.setSecurityManager(new RMISecurityManager());
  }
  try {
    HelloImpl obj = new HelloImpl();
          // Bind this object instance to the name "HelloServer"
    Naming.rebind("//myhost/HelloServer", obj);
    System.out.println("HelloServer bound in registry");
  }
  catch (Exception e) {
    System.out.println("HelloImpl err: " + e.getMessage());
    e.printStackTrace();  } }
```

Hello World in RMI-context - 3b

```
    public void init( ) {
      try { obj = (Hello)Naming.lookup("//" +
                    getCodeBase().getHost() + "/HelloServer");
            message = obj.sayHello();
     } catch (Exception e) {
    System.out.println("HelloApplet exception:"+e.getMessage());
    e.printStackTrace();
    }
  }
public void paint(Graphics g) {
g.drawString(message, 25, 50);
  }
 }
```

Hello World in RMI-context - 3a

```
package examples.hello;                                  // Applet code
 import java.applet.Applet;
 import java.awt.Graphics;
 import java.rmi.Naming;
 import java.rmi.RemoteException;
 public class HelloApplet extends Applet {
   String message = "blank";
// "obj" is the identifier that we'll use to refer
// to the remote object that implements the "Hello" interface

 Hello obj = null;
 public void init() {                                    // see next slide
```

in various languages

Algol Family

Algol-60

```
'BEGIN' 'COMMENT' Hello World in Algol 60;
OUTPUT(4,'("('Hello World!')',/')')'END'
```

Algol-68

```
( # Hello World in Algol 68 # print(("Hello World!",newline)))
```

Assembler-Intel

```
; Hello World for Intel Assembler (MSDOS)

mov ax,cs
mov ds,ax
mov ah,9
mov dx, offset Hello
int 21h
xor ax,ax
int 21h
Hello: db "Hello World!",13,10,"$"
```

Assembler-Linux

```
;; Hello World for the nasm Assembler (Linux)
SECTION .data
        msg db "Hello, world!",0xa ;
         len equ $ - msg
SECTION .text
        global main
main:   mov eax,4       ; write system call
        mov ebx,1       ; file (stdou)
        mov ecx,msg     ; string
        mov edx,len     ; strlen
        int 0x80        ; call kernel
        mov eax,1       ; exit system call
        mov ebx,0
        int 0x80        ; call kernel
```

awk

```
# Hello World in awk
BEGIN {
        print "Hello World!"
        exit
}
```

BrainFxxx

Hello World in BrainF***. No comment character exists.

```
 ++++++++++[>+++++++>++++++++++>+++<<<-
]>++.>+.+++++++ ..+++.>++.<<+++++++++++++++.>.+++.-
-----.-------.>+.
```

C-ANSI

```
/* Hello World in C, Ansi-style */


#include <stdio.h>
#include <stdlib.h>
int main(void) {
        puts("Hello World!");
        return EXIT_SUCCESS;
}
```

C#

```
// Hello World in Microsoft C# ("C-Sharp")

using System;
class HelloWorld

{
        public static int Main(String[] args)

        {
                Console.WriteLine("Hello, World!");
                return 0;
        }

}
```

C++

```
// Hello World in C++ (pre-ISO)

#include <iostream.h> main()

{
        cout << "Hello World!" << endl;
        return 0;
 }
```

```
// Hello World in ISO C++

#include <iostream>
#include <ostream>
 int main() {
        std::cout << "Hello World!" << std::endl;
}
```

Fjölnir

```
;; Hello World in Fjölnir (Icelandic programming language)

"hello" < main
{
   main -> stef(;)
   stofn skrifastreng(;"Halló Veröld!"),
   stofnlok

}
*
"GRUNNUR"
;
```

LaTeX + TeX

LaTeX

```
% Hello World! in LaTeX
\documentclass{article}
\begin{document}
Hello World!
\end{document}
```

TeX

```
% Hello World in plain \TeX
\immediate\write16{Hello World!}
\end
```

Turing Machine

Hello World as a Turing machine.

State	Read	Write	Step	Next state
1	empty	H	>	2
2	empty	e	>	3
3	empty	l	>	4
4	empty	l	>	5
5	empty	o	>	6
6	empty	blank	>	7
7	empty	W	>	8
8	empty	o	>	9
9	empty	r	>	10
10	empty	l	>	11
11	empty	d	>	12
12	empty	!	>	STOP

Cobol

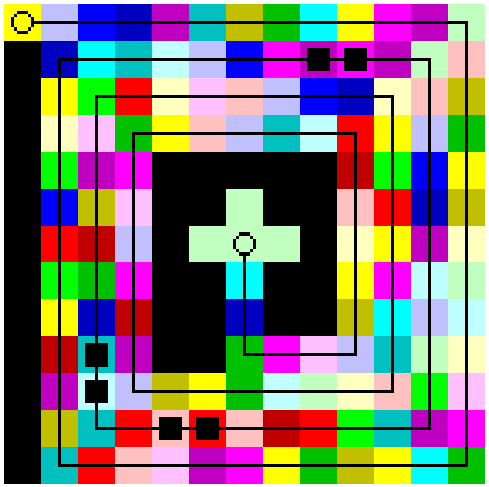
```
* Hello World in Cobol
******************************
IDENTIFICATION DIVISION.
PROGRAM-ID. HELLO.
ENVIRONMENT DIVISION.
DATA DIVISION.
PROCEDURE DIVISION.
MAIN SECTION.
DISPLAY "Hello World!" STOP RUN.
 ******************************
```

Ook

Hello World in Ook. No comments possible.

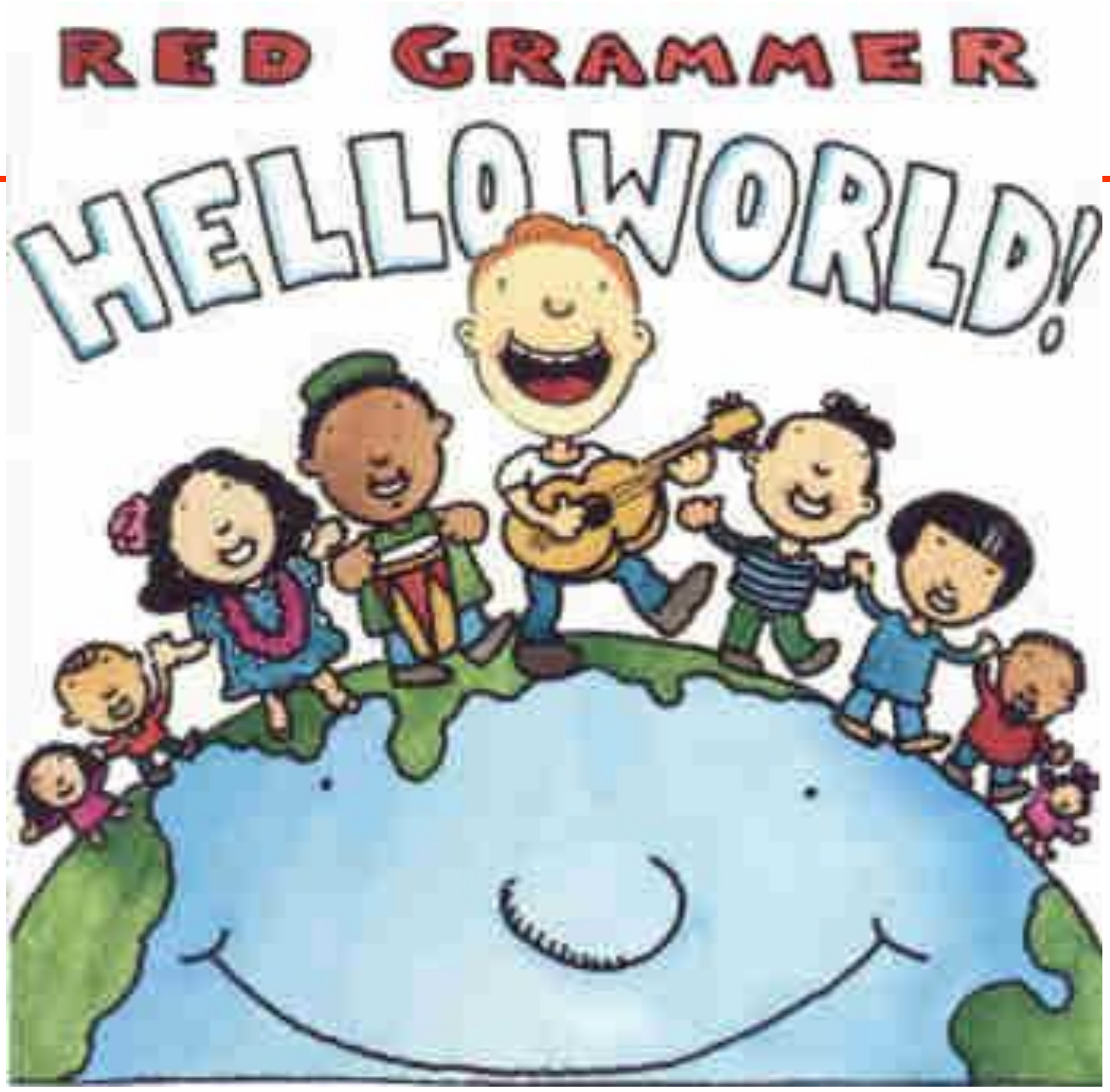
```
Ook. Ook? Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook.
Ook. Ook. Ook. Ook. Ook! Ook? Ook? Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook.
Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook? Ook! Ook! Ook? Ook! Ook? Ook.
Ook! Ook. Ook. Ook? Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook.
Ook. Ook. Ook! Ook? Ook? Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook?
Ook! Ook! Ook? Ook! Ook? Ook. Ook. Ook. Ook! Ook. Ook. Ook. Ook. Ook. Ook. Ook.
Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook! Ook. Ook! Ook. Ook. Ook. Ook. Ook.
Ook. Ook. Ook! Ook. Ook. Ook? Ook. Ook? Ook. Ook? Ook. Ook. Ook. Ook. Ook. Ook.
Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook! Ook? Ook? Ook. Ook. Ook.
Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook? Ook! Ook! Ook? Ook! Ook? Ook. Ook! Ook.
Ook. Ook? Ook. Ook? Ook. Ook? Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook.
Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook! Ook? Ook? Ook. Ook. Ook.
Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook.
Ook. Ook? Ook! Ook! Ook? Ook! Ook? Ook. Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook.
Ook? Ook. Ook? Ook. Ook? Ook. Ook? Ook. Ook! Ook. Ook. Ook. Ook. Ook. Ook. Ook.
Ook! Ook. Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook.
Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook! Ook!
Ook! Ook. Ook. Ook? Ook. Ook? Ook. Ook. Ook! Ook. Ook! Ook? Ook! Ook! Ook? Ook!
Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook.
Ook. Ook. Ook. Ook. Ook! Ook.
```

Hello World in Piet



"Piet" is an esoteric programming language invented by David Morgan-Mar (www.dangermouse.net/esoteric/piet.html).

He writes: "Piet is a programming language in which programs look like abstract paintings. The language is named after Piet Mondrian, who pioneered the field of geometric abstract art."


RED GRAMMER
HELLO WORLD!

Haikus instead of error messages

http://archive.salon.com/21st/chal/1998/02/10chal2.html

Three things are certain:
Death, taxes, and lost data.
Guess which has occurred.

-- David Dixon

- - - - - - - - - - -

Everything is gone;
Your life's work has been destroyed.
Squeeze trigger (yes/no)?

-- David Carlson

- - - - - - - - - - -

I'm sorry, there's -- um --
insufficient -- what's-it-called?
The term eludes me ...

-- Owen Mathews

- - - - - - - - - - -

Windows NT crashed.
I am the Blue Screen of Death.
No one hears your screams.

-- Peter Rothman

- - - - - - - - - - -

Seeing my great fault
Through darkening blue windows
I begin again

-- Chris Walsh

- - - - - - - - - - -

The code was willing,
It considered your request,
But the chips were weak.

-- Barry L. Brumitt

- - - - - - - - - - -

Printer not ready.
Could be a fatal error.
Have a pen handy?

-- Pat Davis

- - - - - - - - - - -

A file that big?
It might be very useful.
But now it is gone.

-- David J. Liszewski

- - - - - - - - - - -

Errors have occurred.
We won't tell you where or why.
Lazy programmers.

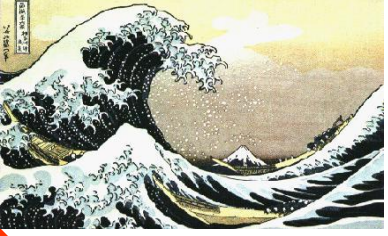
-- Charlie Gibbs

- - - - - - - - - - -

Server's poor response
Not quick enough for browser.
Timed out, plum blossom.

-- Rik Jespersen

- - - - - - - - - - -



Chaos reigns within.
Reflect, repent, and reboot.
Order shall return.

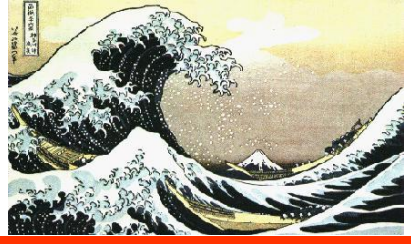
-- Suzie Wagner

- - - - - - - - - - -

Login incorrect.
Only perfect spellers may
enter this system.

-- Jason Axley

- - - - - - - - - - -



This site has been moved.
We'd tell you where, but then we'd
have to delete you.

-- Charles Matthews

- - - - - - - - - - -

wind catches lily
scatt'ring petals to the wind:
segmentation fault

-- Nick Sweeney

- - - - - - - - - - -

ABORTED effort:
Close all that you have.
You ask way too much.

-- Mike Hagler

- - - - - - - - - - -

First snow, then silence.
This thousand dollar screen dies
so beautifully.

-- Simon Firth

- - - - - - - - - - -

With searching comes loss
and the presence of absence:
"My Novel" not found.

-- Howard Korder

- - - - - - - - - - -

The Tao that is seen
Is not the true Tao, until
You bring fresh toner.

-- Bill Torcaso

- - - - - - - - - - -

The Web site you seek
cannot be located but
endless others exist

-- Joy Rothke

- - - - - - - - - - -

Stay the patient course
Of little worth is your ire
The network is down

-- David Ansel

- - - - - - - - - - -

A crash reduces
your expensive computer
to a simple stone.

-- James Lopez

- - - - - - - - - - -

There is a chasm
of carbon and silicon
the software can't bridge

-- Rahul Sonnad

- - - - - - - - - - -

Yesterday it worked
Today it is not working
Windows is like that

-- Margaret Segall

- - - - - - - - - - -

To have no errors
Would be life without meaning
No struggle, no joy

-- Brian M. Porter

- - - - - - - - - - -

You step in the stream,
but the water has moved on.
This page is not here.

-- Cass Whittington

- - - - - - - - - - -

No keyboard present
Hit F1 to continue
Zen engineering?

-- Jim Griffith

- - - - - - - - - - -

Hal, open the file
Hal, open the damn file, Hal
open the, please Hal

-- Jennifer Jo Lane

- - - - - - - - - - -

Out of memory.
We wish to hold the whole sky,
But we never will.

-- Francis Heaney

- - - - - - - - - - -

Having been erased,
The document you're seeking
Must now be retyped.

-- Judy Birmingham

- - - - - - - - - - -

The ten thousand things
How long do any persist?
Netscape, too, has gone.

-- Jason Willoughby

- - - - - - - - - - -

Rather than a beep
Or a rude error message,
These words: "File not found."

-- Len Dvorkin

- - - - - - - - - - -

Serious error.
All shortcuts have disappeared.
Screen. Mind. Both are blank.

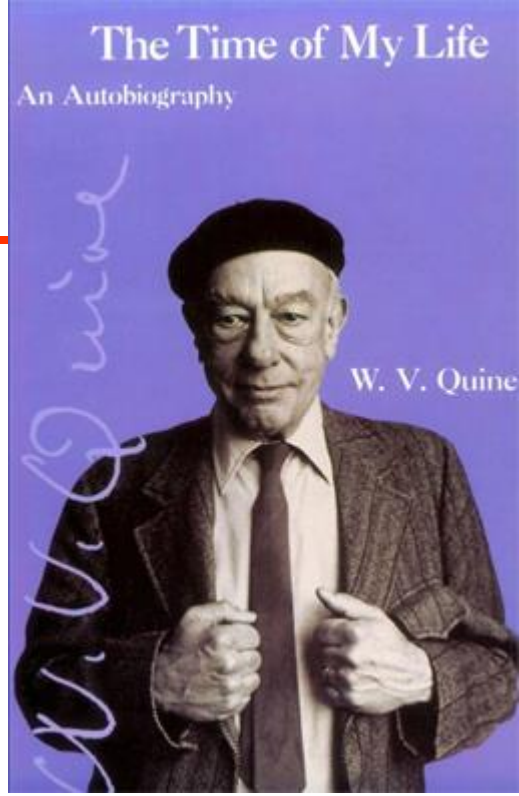
-- Ian Hughes

- - - - - - - - - - -

Self reproducing programs

A program that generates a copy of its own source text as its complete output.

Sources:

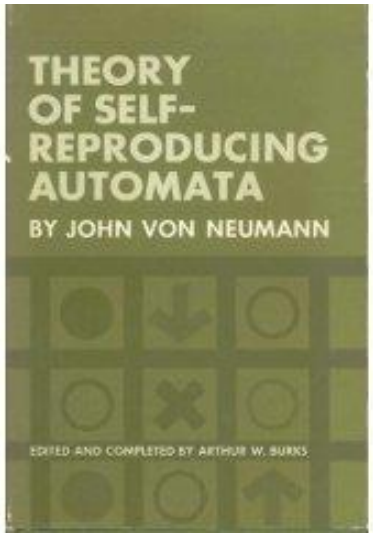


The Quine Page:

http://www.nyx.net/~gthompso/quine.htm

http://www.wvquine.org/

The Theory:



The oldest Quine:

Lisp or Scheme:

```
((lambda (x)
  (list x (list (quote quote) x)))
 (quote
      (lambda (x)
        (list x (list (quote quote) x)))))
```

Classic Quine in 'C'

```
/* newlines may be removed for "better" understanding*/


char*f="char*f=%c%s%c;main()
{printf(f,34,f,34,10);}%c";
main(){printf(f,34,f,34,10);}
```

Quine in Java

Author: Dario Dariol

```
import java.text.*;
class a{public static void main(String x[]){char b[]={34};
char c[]={123};String s[]=new String[3];
s[0]="import java.text.*;
class a{2} public static void main(String x[]){2}char b[]={2}34};
char c[]={2}123}; String s[]=new String[3];s[0]={1}{0}{1};
s[1]=new String(b);s[2]=new String(c);
System.out.println(MessageFormat.format(s[0],s));}}";
s[1]=new String(b);s[2]= new String(c);
System.out.println(MessageFormat.format(s[0],s));}}
```

Quine in Forth

Author: Elko Tchernev (etchernev@acm.org) Note: On some ANS Forths the following self-displaying word will work; I'm not sure if this is cheating or not. (Probably is). :

```
ME S" SEE ME" EVALUATE ;
```

Quine in Perl

Author: Christoph Durr

```
$b='$b=%c%s%c;printf$b,39,$b,39;';printf$b,39,$b,39;
```

Author: Markus Holzer

```
#!/usr/local/bin/perl
$a='#!/usr/local/bin/perl%c$a=%c%s%c;
printf($a,10,39,$a,39,10);%c';printf($a,10,39,$a,39,10);
```

Author: Robin Houston Note: Last line is blank

```
print<<"x2,"\n"
print<<"x2,"\n"
```

Author: Kiriakos Georgiou

```
printf($x,39,$x='printf($x,39,$x=%c%s%c,39);',39);
```

Quine in Perl

Author: Frank Stajano (fstajano@orl.co.uk)

```
l='l=%s;print l%%`l`';print l%`l`
```

Author: Greg Stein (gstein@microsoft.com)

```
x='x=%s\012print x%%`x`' print x%`x`
```

Author: Terry Reedy (tjreedy@udel.edu)

Note: works as an interactive string input. The double quotes could theoretically be removed.

```
"x='x=%s;x%%`x`;x%`x`"
```

Quine in Scheme

Language: Scheme Author: Tanaka Tomoyuki
(tanaka@ucdavis.edu) Note: (Chez Scheme Version 5.0b)

```
(call/cc
  (lambda (c) (c ((lambda (c) `(call/cc (lambda (c) (c (,c ',c)))))
    '(lambda (c) `(call/cc (lambda (c) (c (,c ',c))))))))
```

Author: Tanaka Tomoyuki(tanaka@ucdavis.edu)
Note: (Chez Scheme Version 5.0b)

```
((lambda (q qq) ((lambda (x) `((lambda (q qq) ,(q x)) . ,(q qq)))
  '(lambda (x) `((lambda (q qq) ,(q x)) . ,(q qq)))))
    (lambda (q) `(,q ',q))
      '(lambda (q) `(,q ',q)))
```

Quine in Unix shell

Author:Brian A.E. Meekings (baem@mink.mt.att.com)
Note: sh, ksh

```
b=\\;bb=$b$b$b$b;q=\';x='echo
     "b=$bb;bb=\$b\$b\$b\$b;q=$b$q;x=$q$x$q";echo $x'
echo "b=$bb;bb=\$b\$b\$b\$b;q=$b$q;x=$q$x$q";echo $x
```

Author: Matt Corks njaharve@waterloo.ca
Note: works for bash, ksh, zsh

```
#!/xhbin/bash read foo<<'EOF';eval $foo
echo '#!/bin/bash';echo 'read foo<<""EOF"";eval$foo';
                                        echo $foo;echo EOF
EOF
```

This museum

- The 'vitrinemuseum' shows early computer hardware as used for various labs at Delft University of Technology.

- Have a look at http://vitrinemuseum.ewi.tudelft.nl