1968
@KevlinHenney
Our Reply
31 December 1969
Your feedback will be used to improve Facebook. Thanks for taking the time to make a report.
King Shot to Death in Memphis

Relief Forces Push Close to Khe Sanh

SAIGON (AP)—A U.S. tank force pressed close to the northernmost Viet Cong unit at Khe Sanh on Friday, the fifth day of a drive to lift the siege of the beleaguered fortress in South Vietnam's northwest corner.

At last report, lead elements of the relief force were about 10 miles away from Khe Sanh perimeter and were poised for a frontal assault.

The relief force has faced only light opposition from the enemy and mortar fire since jumping off from Ca Lo, 12 miles away, on Monday. Soviet troops in London said Thursday that the North Vietnamese were beginning to withdraw as a gesture of goodwill, but there was no confirmation from Saigon.

Another reason advanced for the sizable withdrawal was the incessant American bombing of aircraft and artillery in Khe Sanh itself, though, North Vietnamese positions during the past three months.

In other developments:

Chief of Staff: Allied military spokesman in Saigon said the Viet Cong had ordered all U.S. and South Vietnamese military positions Thursday night and early Friday. The order was given, ranging the length of the country, including the big Ton Son-Nui air base on Saigon's outskirts. The heaviest shelling was atStdString Lucid MT Pro

Louisiana Negroses React

By BARRY ZANDER

The death of Dr. Martin Luther King Jr. was "definitely a blow to the Negro race relations" was the opinion of Mr. King's friend, Rev. Dr. Martin Luther King Jr. When he was shot.

Two unidentified men who were arrested were released without charge.

As word of King's death spread through out the city, hordes of people gathered at the site of the shooting on the corner of Figueroa and Wilshire Boulevards. They were released without charge.

The governor of California said he would not say whether the incident was a "hurricane of wrongs."

Satellite Is Now in Wrong Orbit

WASHINGTON (AP)—President Johnson, preparing for a weekend visit to Israel, postponed his trip to Hawaii.

The original plan called for Johnson to also meet in Hawaii with South Korea's president, Chung Ju-Yung. Earlier in the day, Hawaii charged that American planes had violated its airspace.

LBJ Postpones Trip to Hawaii

WASHING'TON (AP)—President Johnson, preparing for a weekend visit to Israel, postponed his trip to Hawaii.

The original plan called for Johnson to also meet in Hawaii with South Korea's president, Chung Ju-Yung. Earlier in the day, Hawaii charged that American planes had violated its airspace.

Governor, HHH Hold Confidential Talk

NEW ORLEANS (AP)—Gov. McKeithen revealed Thursday he had a private, half-hour telephone conversation with Vice President Hubert H. Humphrey, but his discussion was "confidential."
SOFTWARE ENGINEERING

Report on a conference sponsored by the
NATO SCIENCE COMMITTEE
Garmisch, Germany, 7th to 11th October 1968
Software Craftsmanship

The New Imperative

Pete McBreen

Foreword by Dave Thomas
Software engineering was invented to tackle the problems of really large NATO systems projects.
I began to use the term “software engineering” to distinguish it from hardware and other kinds of engineering; yet, treat each type of engineering as part of the overall systems engineering process.

Margaret Hamilton
We must recognize ourselves – not necessarily all of us and not necessarily any one of us all the time – as members of an engineering profession, be it hardware engineering or software engineering.

Anthony A Oettinger
A RESEARCH CENTER FOR AUGMENTING HUMAN INTELLECT

A research center for augmenting human intellect.......................... 395

D. C. Engelbart,
W. K. English
SOFTWARE ENGINEERING

Report on a conference sponsored by the
NATO SCIENCE COMMITTEE
Garmisch, Germany, 7th to 11th October 1968
I've only been seven months with a manufacturer and I'm still bemused by the way they attempt to build software.

Report on a conference sponsored by the NATO SCIENCE COMMITTEE

Garmisch, Germany, 7th to 11th October 2008

JW Smith
They begin with planning specification, go through functional specifications, implementation specifications, etc., etc. This activity is represented by a PERT chart with many nodes.
If you look down the PERT chart you discover that all the nodes on it up until the last one produce nothing but paper.

NATO SCIENCE COMMITTEE

Garmisch, Germany, 7th to 11th October, 1968

J.W. Smith
It is unfortunately true that in my organisation people confuse the menu with the meal.

SOFTWRE ENGINEERING

Report on a conference sponsored by the

NATO SCIENCE COMMITTEE

Garmisch, Germany, 7th to 11th October 1988
The most deadly thing in software is the concept, which almost universally seems to be followed, that you are going to specify what you are going to do, and then do it.
And that is where most of our troubles come from.
The design process is an iterative one.
SOFTWARE ENGINEERING

Conway

unit test

middleware

acceptance testing

layers

iterative

Christopher Alexander

report on a conference sponsored by the NATO science committee, 7th to 11th october 1968

modularity

declarative
Firmitas · Utilitas · Venustas

Marcus Vitruvius Pollio
Patterns Manifesto

We are uncovering better ways of developing software by seeing how others have already done it.
Patterns are an aggressive disregard of originality.

Brian Foote
A capsule definition of engineering, independent of any discipline, as you're likely to find: the set of practices and techniques that have been determined to work reliably through experience.

Glenn Vanderburg
The replication of multiple copies of a software system is the phase of software manufacture which corresponds to the production phase in other areas of engineering.
It is accomplished by simple copying operations, and constitutes only a minute fraction of the cost of software manufacture.

Report on a conference sponsored by the NATO SCIENCE COMMITTEE

Garmisch, Germany, 1-11 October 1979
NOTES ON THE SYNTHESIS OF FORM

CHRISTOPHER ALEXANDER
We may therefore picture the process of form-making as the action of a series of subsystems, all interlinked, yet sufficiently free of one another to adjust independently in a feasible amount of time.
It works, because the cycles of correction and recorrection, which occur during adaptation, are restricted to one subsystem at a time.
circled below, which can, in principle, operate fairly independently.\textsuperscript{32}

We may therefore picture the process of form-making as the action of a series of subsystems, all interlinked, yet sufficiently independent to ensure the emergence of novel forms.
Figure 1
The system, shown at the top, communicates with the outside world through the three interfaces 1, 2, and 3. The middle figure shows the major subsystems, two of which are shown in detail at the bottom.
The basic thesis [...] is that organizations which design systems [...] are constrained to produce designs which are copies of the communication structures of these organizations.

Melvin Conway
How Do Committees Invent?
We have seen that this fact has important implications for the management of system design. [...] A design effort should be organized according to the need for communication.

Melvin Conway
How Do Committees Invent?
I believe that both the total density of information flow necessary between groups, and the percentage of irrelevant information that a given group gets, can be greatly reduced by effectively structuring the object to be constructed and ensuring that this structure is reflected in the structure of the organisation making the product.

Edsger W. Dijkstra
Even though Conway formulated his law around the initial design of a system, the law has important implications for legacy code as well.
μονόλιθος
This is the Unix philosophy: Write programs that do one thing and do it well. Write programs to work together.

Doug McIlroy
µservices
Define a subset of the system which is small enough to bring to an operational state [...] then build on that subsystem.

Report on a conference sponsored by the NATO SCIENCE COMMITTEE

Garmisch, Germany, 7th to 11th October 1988

David
This strategy requires that the system be designed in modules which can be realized, tested, and modified independently, apart from conventions for intermodule communication.
µservices
In the long run every program becomes rococo — then rubble.

Alan Perlis
A software system can best be designed if the testing is interlaced with the designing instead of being used after the design.
bi-quinary coded decimal, *noun*

- A system of representing numbers based on counting in fives, with an additional indicator to show whether the count is in the first or second half of the decimal range, i.e., whether the number represented is in the range 0–4 or 5–9 (or in the range 1–5 or 6–10).
- This system is found in many abacus systems, with paired columns of counters (normally aligned) representing each bi-quinary range.
- The Roman numeral system is also a form of bi-quinary coded decimal.
MCMLXVIII
proc roman numerals = (int year) string:
    skip;
Revised Report on the Algorithmic Language Algol 68

Revised Report on the Algorithmic Language Algol 68

assert (roman numerals (1) = "I")
proc roman numerals = (int year) string:
    "I";
assert (roman numerals (1) = "I");
assert (roman numerals (5) = "V")
proc roman numerals = (int year) string:
    if year = 5 then "V" else "I" fi;
assert (roman numerals (1) = "I");
assert (roman numerals (5) = "V");
assert (roman numerals (10) = "X")
proc roman numerals = (int year) string:
  if year = 10 then "X"
  elsif year = 5 then "V"
  else "I"
  fi;
assert (roman numerals (1) = "I");
assert (roman numerals (5) = "V");
assert (roman numerals (10) = "X");
assert (roman numerals (50) = "L");
assert (roman numerals (100) = "C");
assert (roman numerals (500) = "D");
assert (roman numerals (1000) = "M")
proc roman_numerals = (int year) string:
    if year = 1000 then "M"
    elif year = 500 then "D"
    elif year = 100 then "C"
    elif year = 50 then "L"
    elif year = 10 then "X"
    elif year = 5 then "V"
    else "I"
    fi;
```c
struct (int value, string letters) mapping =
(
    (1000, "M"),
    (500,  "D"),
    (100,  "C"),
    (50,   "L"),
    (10,   "X"),
    (5,    "V"),
    (1,    "I")
);
```
proc roman numerals = (int year) string:
begin
  string result := "";
  for entry from lwb mapping to upb mapping do
    if value of mapping[entry] = year then
      result := letters of mapping[entry]
    fi
  od;
  result
end;
proc roman numerals = (int year) string:
(
    string result := ""
    for entry from lwb mapping to upb mapping do
        if value of mapping[entry] = year then
            result := letters of mapping[entry]
        fi
    od;
    result
);
proposition roman numerals spec =

("Decimal positions correspond to numerals",
((1, "I"), (10, "X"), (100, "C"), (1000, "M"))),

("Quinary values correspond to numerals",
((5, "V"), (50, "L"), (500, "D")))

);
proposition roman numerals spec =
(
("Decimal positions correspond to numerals",
(((1, "I"), (10, "X"), (100, "C"), (1000, "M")))))
("Quinary intervals correspond to numerals",
(((5, "V"), (50, "L"), (500, "D"))))
);

test (roman numerals, roman numerals spec)
mode test = struct (int input, string expected);
mode proposition = struct (string name, flex [1:0] test tests);
proc test = (proc (int) string function, [] proposition spec) void:
    for entry from lwb spec to upb spec do
        print (name of spec[entry]);
        string report := "", separator := " failed:";
        [] test tests = tests of spec[entry];
        for test from lwb tests to upb tests do
            int input = input of tests[test];
            string expected = expected of tests[test];
            string actual = function (input);
            if actual /= expected then
                report +:=
                    separator + " for " + whole (input, 0) +
                    " expected " + expected + " but was " + actual
                separator := ","
            fi
        od;
        print (if report = "" then (new line) else (new line, report, new line) fi) od;
proposition roman numerals spec =

("Decimal positions correspond to numerals",
((1, "I"), (10, "X"), (100, "C"), (1000, "M"))),
("Quinary intervals correspond to numerals",
((5, "V"), (50, "L"), (500, "D"))),
("Multiples of decimals are additive",
((2, "II"), (30, "XXX"), (200, "CC"), (4000, "MMMM")))
);
proc roman numerals = (int year) string:
(
    string result := "";
    int value = year;
    for entry from lwb mapping to upb mapping do
        if value mod value of mapping[entry] = 0 then
            while value > 0 do
                result += letters of mapping[entry];
                value -= value of mapping[entry]
            od
        fi
    od;
    result
);
proposition roman numerals spec =

("Decimal positions correspond to numerals",
 (((1, "I"), (10, "X"), (100, "C"), (1000, "M"))),
("Quinary intervals correspond to numerals",
 (((5, "V"), (50, "L"), (500, "D"))),
("Multiples of decimals are additive",
 (((2, "II"), (30, "XXX"), (200, "CC"), (4000, "MMMM"))),
("Non-multiples of decimals are additive",
 (((6, "VI"), (23, "XXIII"), (273, "CCLXXIII"), (1500, "MD")))
);
proc roman numerals = (int year) string:
(
    string result := "";
    int value = year;
    for entry from lwb mapping to upb mapping do
        while value >= value of mapping[entry] do
            result +:= letters of mapping[entry];
            value -:= value of mapping[entry];
        od
    od;
    result
);
proposition roman numerals spec =
(
  ("Decimal positions correspond to numerals",
   [(1, "I"), (10, "X"), (100, "C"), (1000, "M")]),
  ("Quinary intervals correspond to numerals",
   [(5, "V"), (50, "L"), (500, "D")]),
  ("Multiples of decimals are additive",
   [(2, "II"), (30, "XXX"), (200, "CC"), (4000, "MMMM")]),
  ("Non-multiples of decimals are additive",
   [(6, "VI"), (23, "XXIII"), (273, "CCLXXIII"), (1500, "MD")]),
  ("Numeral predecessors are subtractive",
   [(4, "IV"), (9, "IX"), (40, "XL"), (90, "XC"), (400, "CD"), (900, "CM")])
);
struct (int value, string letters) mapping =
(
    (1000, "M"), (900, "CM"),
    (500,  "D"), (400,  "CD"),
    (100,  "C"), (90,  "XC"),
    (50,  "L"), (40,  "XL"),
    (10,  "X"), (9,  "IX"),
    (5,  "V"), (4,  "IV"),
    (1,  "I"))
);
proposition roman numerals spec =

("Decimal positions correspond to numerals",
  ((1, "I"), (10, "X"), (100, "C"), (1000, "M"))),
("Quinary intervals correspond to numerals",
  ((5, "V"), (50, "L"), (500, "D"))),
("Multiples of decimals are additive",
  ((2, "II"), (30, "XXX"), (200, "CC"), (4000, "MMM"))),
("Non-multiples of decimals are additive",
  ((6, "VI"), (23, "XXIII"), (273, "CCLXXIII"), (1500, "MD"))),
("Numeral predecessors are subtractive",
  ((4, "IV"), (9, "IX"), (40, "XL"), (90, "XC"), (400, "CD"), (900, "CM"))),
("Subtractive predecessors are additive",
  ((14, "XIV"), (42, "XLII"), (1968, "MCMLXVIII")))
);
We instituted a rigorous regression test for all of the features of AWK. Any of the three of us who put in a new feature into the language [...] had to write a test for the new feature.
There is no such question as testing things after the fact with simulation models, but that in effect the testing and the replacement of simulations with modules that are deeper and more detailed goes on with the simulation model controlling, as it were, the place and order in which these things are done.
As design work progresses this simulation will gradually evolve into the real system.

The simulation is the design.
STRUCTURED PROGRAMMING

O. J. DAHL, E. W. DIJKSTRA
and C. A. R. HOARE
goto
Go To Statement Considered Harmful

Key Words And Phrases: go to statement, jump instruction, branch instruction, conditional clause, alternative clause, repetitive clause, program instability, program correctness

Categories: 4.2, 5.20, 5.24

Introduction

For a number of years I have been familiar with the observation that the quality of a program is a decreasing function of the density of go to statements in the program. Indeed, from my recent experience in the most recent version of the language, I believe that the more the go to statements are avoided, the less the potential for misprogramming. And so the program correctness of the language, which is so important for the users, becomes even more crucial. The go to statement is a type of statement which is not only unnecessary, but also dangerous, and I think that it is essential to understand the implications of the go to statement.

My main concern is that the use of go to statements results in a program that is more difficult to understand and maintain. Moreover, it makes the program more prone to errors. For instance, a program that contains many go to statements is more likely to have errors than one that does not. The use of go to statements also makes the program more difficult to test, and it can result in unexpected behavior.

Go to statements are a type of control flow construct that allows a program to jump to a different location in the code. This can be useful in certain situations, but it can also lead to confusion and errors, especially when the program is complex or has many branches. The use of go to statements can make the program more difficult to read and maintain, and it can also lead to unexpected behavior.

Go to statements are not always necessary, and in many cases, they can be avoided by using other control flow constructs, such as conditional statements or loops. By avoiding go to statements, the program becomes more readable and maintainable, and it is less likely to have errors.

In conclusion, I believe that go to statements should be avoided in the language, and that alternative constructs should be used instead. This will result in programs that are more readable, more maintainable, and less prone to errors.

References


snowclone, noun
  ▪ clichéd wording used as a template, typically originating in a single quote
  ▪ e.g., "X considered harmful", "These aren't the Xs you're looking for", "X is the new Y", "It's X, but not as we know it", "No X left behind", "It's Xs all the way down", "All your X are belong to us"
A Case against the GO TO Statement.

by Edsger W. Dijkstra

Technological University

Eindhoven, The Netherlands

Since a number of years I am familiar with the observation that the quality of programmers is a decreasing function of the density of go to statements in the programs they produce. Later I discovered why the use of the go to statement has such disastrous effects and did I become convinced that the go to statement should be abolished from all "higher level" programming languages (i.e. everything except -perhaps- plain machine code). At that time I did not attach too much importance to this discovery; I now submit my considerations for publication because in very recent discussions in which the subject turned up, I have been urged to do so.
Everything should be built top-down, except the first time.

Alan Perlis
You cannot teach beginners top-down programming, because they don't know which end is up.

C A R Hoare
Concept Hierarchies

The construction principle involved is best called *abstraction*; we concentrate on features common to many phenomena, and we abstract away features too far removed from the conceptual level at which we are working.

Ole-Johan Dahl and C A R Hoare
"Hierarchical Program Structures"
Summary—what’s most important.

To put my strongest concerns in a nutshell:

1. We should have some ways of coupling programs like garden hose—screw in another segment when it becomes when it becomes necessary to massage data in another way. This is the way of IO also.

2. Our loader should be able to do link-loading and controlled establishment.

3. Our library filing scheme should allow for rather general indexing, responsibility, generality, data path switching.

4. It should be possible to get private system components (all routines are system components) for bugging around with.

K. D. K. (1962)

Oct. 1st 1964
We should have some ways of coupling programs like garden hose—screw in another segment when it becomes necessary to massage data in another way. This is the way of IO also.
Toutes choses sont dites déjà; mais comme personne n'écoute, il faut toujours recommencer.

André Gide
Everything has been said before; but since nobody listens, we must always start again.

André Gide