N90-14803

INITIAL ADA COMPONENTS EVALUATION

Dr. Travis Moebes
SAIC

SAIC has the responsibility for independent test and validation of the SSE. They have been using a mathematical functions library package implemented in Ada to test the SSE IV&V process. The library package consists of elementary mathematical functions and is both machine and accuracy independent. The SSE Ada components evaluation includes code complexity metrics based on Halstead's software science metrics and McCabe's measure of cyclomatic complexity. Halstead's metrics are based on the number of operators and operands on a logical unit of code and are compiled from the number of distinct operators, distinct operands, and total number of occurrences of operators and operands. These metrics give an indication of the physical size of a program in terms of operators and operands and are used diagnostically to point to potential problems. McCabe's Cyclomatic Complexity Metrics (CCM) are compiled from flow charts transformed to equivalent directed graphs. The CCM is a measure of the total number of linearly independent paths through the code's control structure. These metrics were computed for the Ada mathematical functions library using Software Automated Verification and Validation System (SAVVAS), the SSE IV&V tool. A table with selected results was shown, indicating that most of these routines are of good quality. Thresholds for the Halstead measures indicate poor quality if the length metric exceeds 260 or difficulty is greater than 190. The McCabe CCM indicated a high quality of software products. The SSE will include the Ada version of SAVVAS that may be used for computing these code complexity metrics.

INITIAL ADA COMPONENTS EVALUATION
MATHEMATICAL FUNCTIONS LIBRARY PACKAGE

- o IMPLEMENTED IN ADA BY L. J. GALLAHER, Ph.D
- o LIBRARY PACKAGE OF ELEMENTARY MATHEMATICAL FUNCTIONS
  - SIN, COS, SINH, LOG, ETC.
- o ROUTINES ARE BOTH MACHINE AND ACCURACY INDEPENDENT
  - ACCURACY DETERMINED WHEN LIBRARY PACKAGES ARE INTEGRATED INTO USER PROGRAM

INITIAL ADA COMPONENTS EVALUATION
MATHEMATICAL FUNCTIONS LIBRARY PACKAGE

- o SPECIFICATION FOR ELEMENTARY MATH FUNCTIONS PACKAGE

```
generic type real is digits < >;
package pac_el_fun is --
        function    exp(x : real)         return real;
        function     ln(x : real)         return real;
        function log10(x : real)          return real;
        function   sqrt(x : real)         return real;
        function   cbrt(x : real)         return real;
        function    sin(x : real)         return real;
        function    cos(x : real)         return real;
        function   atan(x : real)         return real;
        function   cosh(x : real)         return real;
        function   sinh(x : real)         return real;
        function    tan(x : real)         return real;
        function   tanh(x : real)         return real;
        function  atanh(x : real)         return real;
        function   asin(x : real)         return real;
        function   acos(x : real)         return real;
        function  asinh(x : real)         return real;
        function  acosh(x : real)         return real;
        function atan2(y, x : real) return real;
        function   "**"(x, y : real) return real;
        function   root(n : integer; y : real) return real;
end pac_el_fun
```

- APPROXIMATIONS CALCULATED DIRECTLY FROM CHEBYSHEV POLYNOMIALS USING A METHOD INTRODUCED BY C. W. CLENSHAW, "CHEBYSHEV SERVICES FOR MATHEMATICAL FUNCTIONS," "NATIONAL PHYSICS LABORATORY MATHEMATICAL TABLES," VOL. 5, HER MAJESTY'S STATIONERY OFFICE, LONDON, 1962

- TECHNICAL REPORT ON THE LIBRARY GIVEN BY L. J. GALLAHER, "A LIBRARY OF ELEMENTARY MATH FUNCTIONS IN ADA," IRAD R626, LOCKHEED-GEORGIA COMPANY, MARIETTA, GEORGIA, JANUARY, 1987

INITIAL ADA COMPONENTS EVALUATION
SSE IV&V CODE COMPLEXITY METRICS

WHAT ARE THE TYPES OF CODE COMPLEXITY METRICS?

- CODE COMPLEXITY METRICS
  - HALSTEAD'S SOFTWARE SCIENCE METRICS
  - McCABE'S MEASURE OF CYCLOMATIC COMPLEXITY

- COMPUTED IN SSE IV&V'S SOFTWARE AUTOMATED VERIFICATION & VALIDATION SYSTEM (SAVVAS)

WHY USE HALSTEAD'S SOFTWARE SCIENCE METRICS?

- COMPILED FROM THE NUMBER OF DISTINCT OPERATORS, DISTINCT OPERANDS, AND TOTAL NUMBER OF OCCURRENCES OF OPERATORS AND OPERANDS
- GIVES AN INDICATION OF THE PHYSICAL SIZE OF A PROGRAM IN TERMS OF OPERATORS AND OPERANDS. VARIOUS SIZE METRICS ARE GIVEN
- USED DIAGNOSTICALLY TO POINT TO POTENTIAL PROBLEMS

- **HALSTEAD'S SOFTWARE SCIENCE METRICS**
  - **BASED ON THE NUMBER OF OPERATORS AND OPERANDS ON A LOGICAL UNIT OF CODE**
  - **OPERATORS INCLUDE ARITHMETIC OPERATORS, BOOLEAN OPERATORS, DELIMITERS**
  - **OPERANDS ARE VARIABLES AND CONSTANTS**
  - **FOR EACH UNIT OF CODE LET:**

    n1 = THE NUMBER OF DISTINCT OPERATORS
    n2 = THE NUMBER OF DISTINCT OPERANDS
    N1 = THE TOTAL NUMBER OF OCCURRENCES OF THE OPERATORS
    N2 = THE TOTAL NUMBER OF OCCURRENCES OF THE OPERANDS

INITIAL ADA COMPONENTS EVALUATION
SSE IV&V CODE COMPLEXITY METRICS

- LENGTH: $N = N1 + N2$
- VOCABULARY: $W = n_1 + n_2$
- VOLUME: $V = N \times \log_2 W$
- LEVEL: $L = (2 \times n_2)/(n_1 \times N_2)$
- DIFFICULTY: $D = 1/L$
- EFFORT: $E = V/L$
- ERROR ESTIMATE: $B = E^{2/3}/E_0; E_0 \cong 3000$

  B = NO. OF BUGS IN A PROGRAM

- INTERPRETATION OF HALSTEAD'S METRICS

  THE LENGTH N SERVES AS A MEASURE OF MODULARITY. A LENGTH OF GREATER THAN 260 INDICATES POOR QUALITY CODE. THE CODE SHOULD (PROBABLY) BE REDUCED TO MORE AND SMALLER MODULES

  THE VOLUME V REPRESENTS THE SIZE (IN BITS) OF A LOGICAL UNIT OF CODE

  THE LEVEL L IS A MEASURE THAT RELATES TO THE EFFORT OF WRITING, PROPENSITY (INCLINATION) FOR ERROR, AND EASE OF UNDERSTANDING OF A LOGICAL UNIT OF CODE

INITIAL ADA COMPONENTS EVALUATION
SSE IV&V CODE COMPLEXITY METRICS

THE DIFFICULTY D, THE RECIPROCAL OF LEVEL, INDICATES THE DIFFICULTY IN UNDERSTANDING AND MAINTAINING THE CODE. A DIFFICULTY GREATER THAN 190 TENDS TO INDICATE A POOR QUALITY OF CODE

THE EFFORT E IS A MEASURE OF THE RELATIVE AMOUNT OF WORK INVOLVED IN PRODUCING A PIECE OF CODE

THE ERROR ESTIMATE B (BUGS) IS THE ESTIMATED NUMBER OF ERRORS IN THE CODE.

Operator Parameters

Operator	j	$f_{1,j}$
;	1	9
:=	2	6
( ) or BEGIN...END	3	5
IF	4	3
=	5	3
/	6	1
-	7	1
x	8	1
RETURN	9	1
EXIT	10	1
	$n_1=10$	$N_1=31$

INITIAL ADA COMPONENTS EVALUATION
SSE IV&V CODE COMPLEXITY METRICS

Operand Parameters

Operand	j	$f_{2,j}$
B	1	6
A	2	5
O	3	3
R	4	3
G	5	2
GOD	6	2
	$n_2=6$	$N_2=21$

WHY USE McCABE'S CYCLOMATIC COMPLEXITY METRICS?

- o COMPILED FROM FLOWCHARTS TRANSFORMED TO EQUIVALENT DIRECTED GRAPHS, THE NUMBER OF EDGES OF THE GRAPHS, THE NUMBER OF NODES OF THE GRAPHS AND THE NUMBER OF SEPARATE PARTS OF THE GRAPH
- o THE McCABE CYCLOMATIC COMPLEXITY METRIC (CCM) ASSISTS IN BREAKING UP A SOFTWARE PROGRAM TO COMPONENTS THAT HAVE A SMALL CCM <10
- o PROGRAMS WITH LARGE CCM SHOULD HAVE MORE ERRORS DURING DEVELOPMENT

WHY USE McCABE'S CYCLOMATIC COMPLEXITY METRICS? (Continued)

- o THE CCM MAY BE USED TO DETERMINE THE MINIMUM NUMBER OF PATHS NEEDED.
- o THE CCM RELATES ONLY TO LOGICAL COMPLEXITY. THE CCM SHOULD BE USED IN CONJUNCTION WITH OTHER METRICS

McCABE'S CYCLOMATIC COMPLEXITY METRIC

McCABE'S CYCLOMATIC COMPLEXITY METRIC IS A MEASURE OF THE TOTAL NUMBER OF LINEARLY INDEPENDENT PATHS THOUGH THE CODES CONTROL STRUCTURE

TO CALCULATE CYCLOMATIC COMPLEXITY, FLOW-CHARTS ARE TRANSFORMED TO EQUIVALENT DIRECTED GRAPHS

McCABE'S CYCLOMATIC COMPLEXITY METRIC (Continued)

COMPLEXITY MEASURE OF A PROGRAM IS A FUNCTION OF THE NUMBER OF DECISIONS IN A PROGRAM AND IS GIVEN BY A SINGLE NUMBER KNOWN AS A CYCLOMATIC NUMBER

COMPLEXITY MEASURE IS INDEPENDENT OF THE PHYSICAL SIZE OF THE PROGRAM

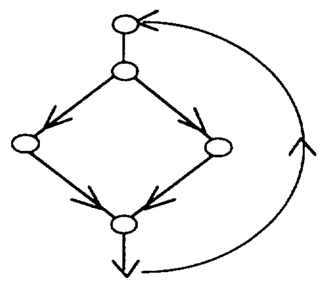
DEFINITIONS:

A GRAPH IS A TREE STRUCTURE CONSISTING OF NODES CONNECTED BY BRANCHES

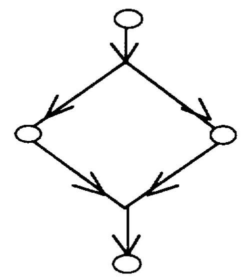
A DIRECTED GRAPH IS A GRAPH IN WHICH A DIRECTION OR FLOW IS ASSOCIATED WITH EVERY BRANCH

DEFINITIONS: (Continued)

A STRONGLY CONNECTED GRAPH IS A GRAPH THAT HAS A UNIQUE ENTRY AND EXIT NODE AND EACH NODE CAN BE REACHED FROM EVERY OTHER NODE

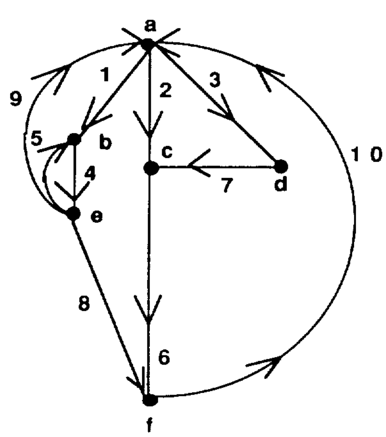


Strongly Connected



Not Strongly Connected

EXAMPLE
CONTROL GRAPH G



SOME POSSIBLE PATHS

a b e f

b e b

a b e a

a c f a

a d c f a

a b e f a 2b e b a b e a

DEFINITION:

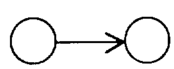
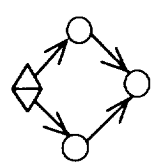
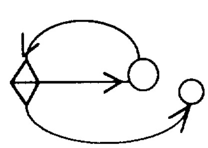
THE CYCLOMATIC NUMBER V(G) OF A GRAPH G WITH n NODES AND e BRANCHES AND p CONNECTED COMPONENTS IS

V(G)=e-n+2P;

IN THE EXAMPLE CONTROL GRAPH G:

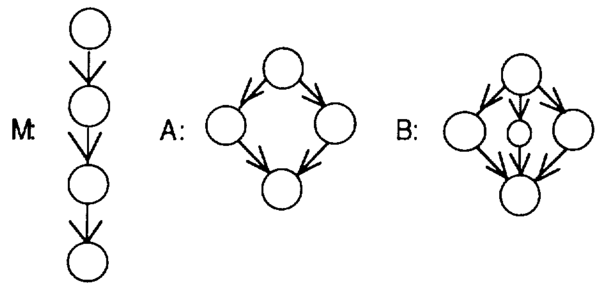
NUMBER OF NODES = 6
NUMBER OF BRANCHES = 10
V(G) = 10-6+2(1) = 6

GRAPH EXAMPLES:

CONTROL	STRUCTURE	CYCLOMATIC COMPLEXITY V = e - n + 2p
SEQUENCE		V = 1- 2 + 2 = 1
IF THEN ELSE		V = 4 - 4 + 2 = 2
WHILE		V = 3 - 3 + 2 = 2

GRAPH EXAMPLE WITH p = NO. OF COMPONENTS = 3

SUPPOSE A PROGRAM M AND TWO CALLED SUBPROGRAMS A AND B HAVE THE FOLLOWING CONTROL STRUCTURE:



THEN G = MUAUB AND p = 3
THEN V(G) = V(MUAUB) = 13-13+2x3 = 6
NOTE: V(MUAUB) = V(M)+V(A)+V(B)

- PROPERTIES OF THE CYCLOMATIC COMPLEXITY

1. V(G)≥1

2. V(G) IS THE MAXIMUM NUMBER OF (LINEARLY) INDEPENDENT PATHS IN G; IT IS THE SIZE OF A BASIS SET (i.e., ALL COMBINATIONS OF PATHS IN THE CODE ARE MADE UP FROM PATHS IN G)

3. INSERTING OR DELETING FUNCTIONAL STATEMENTS TO G DOES NOT AFFECT V(G)

4. G HAS ONLY ONE PATH IF AND ONLY IF V(G)=1

PROPERTIES OF THE CYCLOMATIC COMPLEXITY (Continued)

5. INSERTING A NEW EDGE IN G INCREASES V(G) BY UNITY

6. V(G) DEPENDS ONLY ON THE DECISION STRUCTURE OF G

   V(G) ASSISTS IN BREAKING UP A SOFTWARE PROGRAM TO COMPONENTS THAT HAVE A SMALLER COMPLEXITY NUMBER. McCABE RECOMMENDS THAT EACH COMPONENT G HAS A V(G) LESS THAN 10

SAVVAS-ADA-McCABE'S CCM

o COUNT THE NUMBER OF "CASE" STATEMENT BRANCHES AND SET THIS NUMBER EQUAL TO m_1

```
CASE TODAY IS

1 )  WHEN MON        =>  OPEN_ACCOUNTS;
                         COMPUTE_INITIAL BALANCE;
1 )  WHEN TUE..THU   =>  GENERATE_REPORT (TODAY);
1 )  WHEN FRI        =>  COMPUTE_CLOSING_BALANCE;
                     =>  CLOSE_ACCOUNTS:
1 )  WHEN SAT/SUN    =>  NULL
—
4    END CASE;
                   4 BRANCHES
```

- o COUNT THE NUMBER OF "IF STATEMENT" BRANCHES AND SET THIS NUMBER TO m_2

```
1 ) IF WEATHER_CONDITION = RAIN THEN
        COMPUTE_RAINFALL;
1 ) ELSIF WEATHER_CONDITION = SUNSHINE THEN
        COMPUTE_HUMIDITY:
1 ) ELSE
        COMPUTE_PRES;
    END IF;
___
3
```

3 BRANCHES

- o COUNT THE NUMBER OF "LOOP BRANCHES" AND SET THIS NUMBER TO m_3

```
1 ) FOR I IN 1..10 LOOP
1 )     FOR J IN 1..20 LOOP
            IF A(I,J) = 0 THEN
                M:=I ;
                N:=J;
                EXIT FIND;
            END IF;
        END LOOP;
    END LOOP FIND;
```

2 LOOP BRANCHES 1 "IF STATEMENT" BRANCHES

3 BRANCHES

$CCM=LOG_2\ m_1+m_2+m_3$

INITIAL ADA COMPONENTS EVALUATION SAVVAS CCM RESULTS APPLIED TO ADA MATHEMATICAL FUNCTIONS LIBRARY

UNIT NAME	McCABE V(g)	HALSTEAD's Length Metric (260)	HALSTEAD's Difficulty Metric (190)
Average	3	82	16.018
ln	1	32	5.25
tanh	3	74	16.7
t_aux_fun	1	163	25.84
pac_aux_fun	1	215	32.045
remainder	4	90	34.833
pac_el_fun	1	868	19.989

INITIAL ADA COMPONENTS EVALUATION SUMMARY

- o CCM DIRECTED IV&V APPLIED TO THE SSE ADA MATHEMATICAL FUNCTION LIBRARY PACKAGE INDICATED A HIGH QUALITY OF SOFTWARE PRODUCTS
- o IN HALSTEAD'S SOFTWARE SCIENCE METRICS, A LENGTH OF GREATER THEN 260 OR DIFFICULTY GREATER THAN 190 TENDS TO INDICATE POOR QUALITY CODE
- o McCABE HAS SUGGESTED THAT A CYCLOMATIC COMPLEXITY OF 10 SHOULD BE THE UPPER LIMIT FOR V(G) (CYCLOMATIC COMPLEXITY MEASURE FOR A GRAPH G)
- o THE ADA VERSION OF SAVVAS MAY BE USED FOR COMPUTING CODE COMPLEXITY METRICS