



Essential Metrics
Project Manager (EPM)
for ADA, C#, C/C++, Java, VB6 / VB.NET,
Perl and PL/SQL

Proof of Results: Ada
1.10.000

Contents

Contents	2
Document	2
Purpose	2
Readership.....	2
Results	3

Document

Purpose

This document is a Proof of Metrics Results for the Essential Project Manager source code metrics tool.

Readership

This User Guide is intended for end-users of the source code metrics results. Management and non-technical users should refer to our website – <http://www.powersoftware.com/epm/>

Results

Project Metrics for Ada_Proof_b (15/03/2006)

Metric	Old	New	Diff
LOC	107	114	+7
SLOC	58	65	+7
NSC	40	48	+8
N1	64	64	
N2	98	118	+20
n1	22	20	-2
n2	33	34	+1
N	162	182	+20
n	55	54	-1
V	936	1047	+111
D	32	34	+2
E (k)	29	35	+6
B	0	0	+0
J_COM	0	0	
C_COM	0	0	
CPP_COM	19	21	+2
COM_LOC	19	21	+2
BYTES	2950	3227	+277
NFILE	1	1	

Changed Metrics

Metric	Value
CHG_SLOC	8
DEL_SLOC	1
ADD_SLOC	8
CHG_FILE	1
DEL_FILE	0
ADD_FILE	0

1	-- =====	1	-- =====		
2	--	2	--		
3	-- TASK NAME: numbers.search.process_normal_input	3	-- TASK NAME: numbers.search.process_normal_input		
4	--	4	--		
5	-- ALGORITHM/STRATEGY: Search parameters are read from file and, for each	5	-- ALGORITHM/STRATEGY: Search parameters are read from file and, for each		
6	-- set, entry do_search.enter_start_search is called.	6	-- set, entry do_search.enter_start_search is called.		
7	--	7	--		
8	-- NOTES: None.	8	-- NOTES: None.		
9	--	9	--		
10	-- =====	10	-- =====		
11		11			
12	WITH other_io;	12	WITH other_io;		
13	WITH text_io;	13	WITH text_io;		
14		14			
15	SEPARATE (numbers.search)	15	SEPARATE (numbers.search)		
16		16			
17	TASK BODY process_normal_input IS	17	TASK BODY process_normal_input IS		
18		18			
19	-- Input file	19	-- Input file		
20	input : text_io.file_type;	20	input_file : text_io.file_type;		
21	-- Order of PDIs to be sought	21	-- Order of PDIs to be sought		
22	order : natural;	22	order : natural;		
23	-- Radix of PDIs to be sought	23	-- Radix of PDIs to be sought		
24	radix : numbers.radix_range;	24	radix : numbers.radix_range;		
25	-- Length of PDIs to be sought	25	-- Length of PDIs to be sought		
26	search_length : numbers.digit_range;	26	search_length : numbers.digit_range;		
27		27			
28	-- Exception raised for any error related to opening input file	28	-- Exception raised for any error related to opening input file		
29	file_open_error : EXCEPTION;	29	file_open_error : EXCEPTION;		
30					
31	BEGIN	31	BEGIN -- process_normal_input		
32		32			
33	SELECT	33	SELECT		
34		34			
		35	-- Wait for signal to begin reading file		
35	ACCEPT start;	36	ACCEPT start;		
36		37			

37	BEGIN	38	BEGIN
38		39	
39	-- Open input file	40	-- Open input file
40	text_io.open (file => input, mode => text_io.in_file,	41	text_io.open (file => input_file, mode => text_io.in_file,
41	name => "input.dat");	42	name => "input.dat");
42		43	
43	EXCEPTION	44	EXCEPTION
44		45	
45	WHEN text_io.name_error =>	46	WHEN text_io.name_error =>
46	text_io.put_line ("Cannot find file ""input.dat""");	47	text_io.put_line ("Cannot find file ""input.dat""");
47	RAISE file_open_error;	48	RAISE file_open_error;
48		49	
49	WHEN text_io.use_error =>	50	WHEN text_io.use_error =>
50	text_io.put_line ("Cannot open file ""input.dat""");	51	text_io.put_line ("Cannot open file ""input.dat""");
51	RAISE file_open_error;	52	RAISE file_open_error;
52		53	
53	WHEN OTHERS =>	54	WHEN OTHERS =>
54	text_io.put_line ("Problem encountered opening file " &	55	text_io.put_line ("Problem encountered opening file " &
55	""input.dat""");	56	""input.dat""");
56	RAISE file_open_error;	57	RAISE file_open_error;
57		58	
58	END;	59	END;
59		60	
60	-- Process search parameters until end-of-file reached	61	-- Process search parameters until end-of-file reached
61	WHILE (NOT text_io.end_of_file(input)) LOOP	62	WHILE (NOT text_io.end_of_file(input_file)) LOOP
62		63	
63	-- Read parameters	64	-- Read parameters
64	other_io.natural_io.get(input, radix);	65	other_io.natural_io.get(input_file, radix);
65	other_io.natural_io.get(input, order);	66	other_io.natural_io.get(input_file, order);
66	other_io.natural_io.get(input, search_length);	67	other_io.natural_io.get(input_file, search_length);
67	text_io.skip_line(file => input);	68	text_io.skip_line(file => input_file);
68		69	