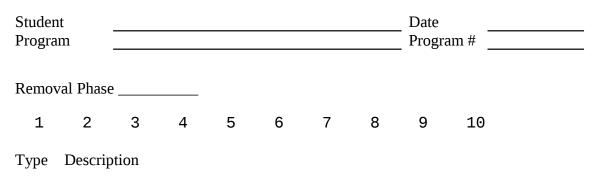
Defect Types		
10 Syntax	50 Architecture	90 Build, Package
20 Assignment	60 Data	100 Environment
30 Algorithm	70 Checking	110 System
40 Interface	80 Documentation	·

Tally Marks by Inject Phase				
-	Design	X Compile		
/	Code	🛎 Test		
0	Code Review	1		

Defect Tally



Remov	val Phase	<u> </u>							
1	2	3	4	5	6	7	8	9	10
Type	Descrip	otion							

© Copyright 2002 by Redpoint Research

PSQ Defect Type Standard

Type Number	Type Name	Description
10	Syntax	Spelling, punctuation, typos, instruction formats
20	Assignment	Declaration, duplicate names, scope, data range, initialization of data.
30	Algorithm	Errors in algorithm design; logic, pointers, loops, recursion, computation
40	Interface	Errors in module interface design: procedure calls and references, parameter lists.
50	Architecture	Errors in architectural design; modularization, structure, coupling, cohesion.
60	Data	Errors in data design: structure, content
70	Checking	Failure to properly validate data values before used; error messages, asserts.
80	Documentation	Source code comments, messages. Also external documentation.
90	Build, Package	change management, library, version control, makefile error, etc.
100	Environment	CASE tool, compiler, test, or other support system problems.
110	System	Hardware and platform configuration, real-time resources, shared memory.

Defect Tally Instructions

Purpose	This form holds a tally of defects you find during development.			
1 urpose	Use these data to complete the Project Plan Summary.			
General	Record all defects during unit development in this log.A defect is any change you made to the source code.Don't count error messages; Establish the underlying defect in the source code. These defects are NOT entered in the team's defect tracking system.If you need additional space, use another copy of the form. It is best to fill out the forms in pen or pencil. Don't bother to type them for submission.			
Header	 Enter the following: your name today's date the program or module name the number of the PSQ program 			
Phase	Enter the phase during which the defect was removed . This would generally be the phase during which you found and fixed the defect.			
Tally	Keep track of the number of defects found by marking off a number in the list for each defect. Use your best judgment to determine the phase where the defect was injected and use the corresponding kind of tally mark shown below.			
	Mark Phase Injected			
	— Design / Code O Code Review X Compile ☎ Test			
Notes	You may record notes about the defects you find. The kinds of data that might be useful would be data that normally is entered into a Defect Recording Log: defect type, description of defect, and fix time.			

Example Defect Tally

Delesi Types M Aynon 50 Artiklavikus: 96 Juliki, Perkage M Angemeres 60 Data 107 Deta 107 Deta 40 Angemeren /2 Cherking 118 System 20 Interier: 20 Degenermados	Tally Marks by Inject Phase - Design & Compile / Code D Dest O Dode Review					
Defect Tally						
Student Joe Programmer	•		Date	10/7/01		
Program Romen Numeral	Conve	SIM	Program	a# (
Removal Phase						
	7	8	9	10		
Type Description						
30 missing log increment						
Removal Phase <u>COMPLE</u> + / / 4 5 6 Type Description 40 wrong return type 10 missing brace 10 missing semicolon	7	Ħ	'n	10		
Removal Phase TEST						
- <u>+</u> / 3 4 5 6	7	8	9	10		
Type Description						
20 forgot to initialize sum to) tero					
20 forgot to initialize sum to 10 "== " should be ".equals	* fr	5tm	p			

© Copyright 2002 by Redpoint Research