Assignment Kit for Program 4

Personal Software Process (PSP) for Engineers: Part I

The Software Engineering Institute (SEI)
is a federally funded research and development center
sponsored by the U.S. Department of Defense and
operated by Carnegie Mellon University.

Personal Software Process for Engineers: Part I

Assignment Kit for Program 4

Overview

Overview

This assignment kit covers the following topics.

Prerequisites

Reading
- Chapter 7

Program 4 requirements

Program 4 requirements

Using PSP1.1, write a program to calculate relative size ranges for very small, small, medium, large, and very large ranges using standard deviation.

Thoroughly test the program. Test the program using the data provided in tables 1 and 2. Expected values are included in table 3.

Class Name	Class LOC	Number of Methods
each_char	18	3
string_read	18	3
single_character	25	3
each_line	31	3
single_char	37	3
string_builder	82	5
string_manager	82	4
list_clump	87	4
list_clip	89	4
string_decrementer	230	10
Char	85	3
Character	87	3
Converter	558	10

Table 1. LOC/Method Data

Chapter	Pages
Preface	7
Chapter 1	12
Chapter 2	10
Chapter 3	12
Chapter 4	10
Chapter 5	12
Chapter 6	12
Chapter 7	12
Chapter 8	12
Chapter 9	8
Appendix A	8
Appendix B	8
Appendix C	20
Appendix D	14
Appendix E	18
Appendix F	12

Table 2. Pgs/Chapter

Continued on next page

Program 4 requirements, continued

	VS	S	M	L	VL
LOC/Method	4.3953	8.5081	16.4696	31.8811	61.7137
Pgs/Chapter	6.3375	8.4393	11.2381	14.9650	19.9280

Table 3. Expected Values

Relative Size Table

Using relative size tables in the PSP

In the PSP, relative size tables are used to give you a framework for judging the size of new parts in your planned products. For example, if you know the sizes of all previously developed parts of a certain type, you can then better judge the likely size of a new part of that type. The standard deviation procedure described in the following section allows you to balance your estimates so they more or less conform to the normal distribution.

The medium range (M) is the area from -0.5 standard deviations to +0.5 standard deviations from the mean, as shown in Figure 1. Assuming that the data approximates a normal distribution, the likely number of parts that are within plus or minus 0.5 standard deviation of the average value is 38.3 percent. Following similar logic, the range percentages area are as follows:

- 6.68 % should be very small
- 24.17% should be small
- 38.2% should be medium
- 24.17% should be large
- 6.68% should be very large

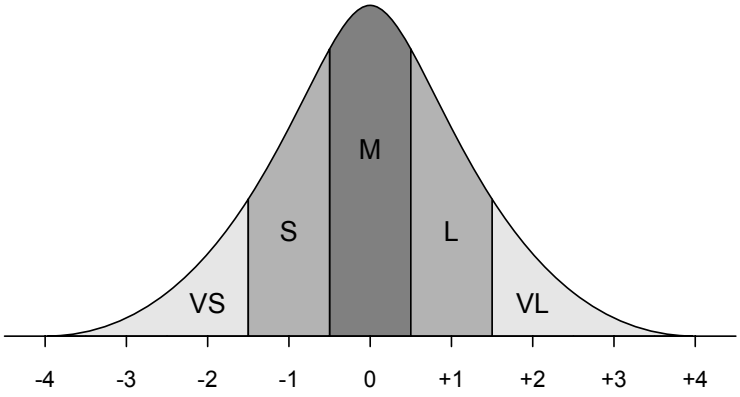

Figure 1. Ranges of standard deviations

Calculating a relative size table using standard deviation

The PROBE estimating method divides historical size data into categories that represent your kind of work. One way of doing this is based on standard deviation. First, divide your historical data into functional categories that each have at least 6 to 8 members (calculation, text, and data, for example). For each category you can then calculate the relative size ranges for VS, S, M, L, and VL following the below procedure.

1. Divide the part sizes by the number of items in each part to determine size per item, if applicable. For instance, you may not have enough data on classes to develop a relative size table, but you do have sufficient data on methods. Instead of using total LOC per class, you can use LOC/method.

Continued on next page

Calculating a relative size table using standard deviation, continued

2. Next, you'll need to log-normally transform the data. This is necessary because you cannot have negative sizes and the smaller values tend to bunch up. Log-normally transforming the data allows you plot the data around a mean of zero. For each size value, x_i, take the natural logarithm, ln, to give ln (x_i).

3. Calculate the average of these *n* logarithmic values: $avg = \frac{\sum_{i=1}^{n} \ln(x_i)}{n}$.

4. Calculate the variance of these values: $\text{var} = \sigma^2 = \frac{\sum_{i=1}^{n} (\ln(x_i) - avg)^2}{(n-1)}$.

5. Calculate the standard deviation: $\sigma = \sqrt{\text{var}}$.

6. Calculate the logarithmic ranges:

   ln(VS) = $avg - 2\sigma$

   ln(S) = $avg - \sigma$

   ln(M) = avg

   ln(L) = $avg + \sigma$

   ln(VL) = $avg + 2\sigma$

7. Last, convert the natural log values back to their original form by calculating the anti-logarithm (calculate *e* to the power of the log value) to get the midpoints of the size ranges

   VS = $e^{\ln(VS)}$

   S = $e^{\ln(S)}$

   M = $e^{\ln(M)}$

   L = $e^{\ln(L)}$

   VL = $e^{\ln(VL)}$

Example of calculating a relative size table

Example of calculating a relative size table

In this example, we'll calculate the relative size ranges for very small, small, medium, large, and very large ranges using standard deviation for the data in the table below.

Class Name	Class LOC	Number of Methods	LOC/method
each_char	18	3	6.0000
string_read	18	3	6.0000
single_character	25	3	8.3333
each_line	31	3	10.3333
single_char	37	3	12.3333
string_builder	82	5	16.4000
string_manager	82	4	20.5000
list_clump	87	4	21.7500
list_clip	89	4	22.2500
string_decrementer	230	10	23.0000
Char	85	3	28.3333
Character	87	3	29.0000
Converter	558	10	55.8000

1. Divide the part sizes by the number of items in each part to determine size per item, if applicable. In this instance, the LOC/method is calculated for each class.
2. For each size value, x_i, calculate the natural logarithm, ln, to give ln(x_i).

Class Name	LOC/method	ln(x_i)
each_char	6.0000	1.7918
string_read	6.0000	1.7918
single_character	8.3333	2.1203
each_line	10.3333	2.3354
single_char	12.3333	2.5123
string_builder	16.4000	2.7973
string_manager	20.5000	3.0204
list_clump	21.7500	3.0796
list_clip	22.2500	3.1023
string_decrementer	23.0000	3.1355
Char	28.3333	3.3440
Character	29.0000	3.3673
Converter	55.8000	4.0218
Total		36.4197

Continued on next page

Example of calculating a relative size table, Continued

Example of calculating a relative size table, continued

3. Calculate the average of these *n* logarithmic values:

$$avg = \frac{\sum_{i=1}^{n} \ln(x_i)}{n} = \frac{36.4197}{13} = 2.8015.$$

4. Calculate the variance of these values:

$$\text{var} = \sigma^2 = \frac{\sum_{i=1}^{n} \left(\ln(x_i) - avg\right)^2}{(n-1)} = \frac{5.2350}{12} = 0.4363$$

Class Name	LOC/method	$ln(x_i)$	$(ln(x_i)-avg)^2$
each_char	6.0000	1.7918	1.0196
string_read	6.0000	1.7918	1.0196
single_character	8.3333	2.1203	0.4641
each_line	10.3333	2.3354	0.2173
single_char	12.3333	2.5123	0.0836
string_builder	16.4000	2.7973	0.0000
string_manager	20.5000	3.0204	0.0479
list_clump	21.7500	3.0796	0.0773
list_clip	22.2500	3.1023	0.0905
string_decrementer	23.0000	3.1355	0.1115
Char	28.3333	3.3440	0.2943
Character	29.0000	3.3673	0.3201
Converter	55.8000	4.0218	1.4890
Total		36.4197	5.2350

5. Calculate the standard deviation: $\sigma = \sqrt{\text{var}} = \sqrt{0.4363} = 0.6605$
6. Calculate the logarithmic ranges:

   ln(VS) = $avg - 2\sigma = 2.8015 - 1.3210 = 1.4805$

   ln(S) = $avg - \sigma = 2.8015 - 0.6605 = 2.1410$

   ln(M) = $avg = 2.8015$

   ln(L) = $avg + \sigma = 2.8015 - 0.6605 = 3.4620$

   ln(VL) = $avg + 2\sigma = 2.8015 - 1.3210 = 4.1225$

Continued on next page

Example of calculating a relative size table, Continued

Example of calculating a relative size table, continued

7. Convert the natural log values back to their original form by calculating the anti-logarithm by calculating e to the power of the log value to determine the midpoints of the size ranges:

   VS = $e^{\ln(VS)} = e^{1.4805} = 4.3953$

   S = $e^{\ln(S)} = e^{2.1410} = 8.5081$

   M = $e^{\ln(M)} = e^{2.8015} = 16.4696$

   L = $e^{\ln(L)} = e^{3.4620} = 31.8811$

   VL = $e^{\ln(VL)} = e^{4.1225} = 61.7137$

Assignment instructions

Assignment instructions

Before starting program 4, review the top-level PSP1.1 process script below to ensure that you understand the "big picture" before you begin. Also, ensure that you have all of the required inputs before you begin the planning phase.

PSP1.1 Process Script

Purpose	To guide the development of module-level programs
Entry Criteria	- Problem description - PSP1.1 Project Plan Summary form - Size Estimating template - Historical size and time data (estimated and actual) - Time and Defect Recording logs - Defect Type, Coding, and Size Counting standards - Stopwatch (optional)

Step	Activities	Description
1	Planning	- Produce or obtain a requirements statement. - Use the PROBE method to estimate the added and modified size of this program. - Complete the Size Estimating template. - Use the PROBE method to estimate the required development time. - ***Complete a Task Planning template.*** - ***Complete a Schedule Planning template.*** - Enter the plan data in the Project Plan Summary form. - Complete the Time Recording log.
2	Development	- Design the program. - Implement the design. - Compile the program, and fix and log all defects found. - Test the program, and fix and log all defects found. - Complete the Time Recording log.
3	Postmortem	Complete the Project Plan Summary form with actual time, defect, and size data.

Exit Criteria	- A thoroughly tested program - Completed Project Plan Summary form with estimated and actual data - Completed Size Estimating ***and Task and Schedule Planning*** templates - Completed Test Report template - Completed PIP forms - Completed Time and Defect Recording logs

Continued on next page

Planning phase

Plan program 4 following the PSP1.1 planning phase and the PROBE estimating scripts.

PSP1.1 Planning Script

Purpose	To guide the PSP planning process
Entry Criteria	- Problem description - PSP1.1 Project Plan Summary form - Size Estimating, ***Task Planning, and Schedule Planning templates*** - Historical size and time data (estimated and actual) - Time Recording log

Step	**Activities**	**Description**
1	Program Requirements	- Produce or obtain a requirements statement for the program. - Ensure that the requirements statement is clear and unambiguous. - Resolve any questions.
2	Size Estimate	- Produce a program conceptual design. - Use the PROBE method to estimate the added and modified size of this program. - Complete the Size Estimating template and Project Plan Summary form.
3	Resource Estimate	- Use the PROBE method to estimate the time required to develop this program. - Using the To Date % from the most recently developed program as a guide, distribute the development time over the planned project phases. (Note: This step is completed by the SEI student workbook.)
4	***Task and Schedule Planning***	***For projects lasting several days or more, complete the Task Planning and Schedule Planning templates.***

Exit Criteria	- Documented requirements statement - Program conceptual design - Completed Size Estimating template - ***For projects lasting several days or more, completed Task and Schedule Planning templates*** - Completed Project Plan Summary form with estimated program size and development time data - Completed Time Recording log

Verify that you have met all of the exit criteria for the planning phase, **then have an instructor review your plan**. After your plan has been reviewed, proceed to the development phase.

Continued on next page

Use the PROBE method to create size and resource estimates.

PROBE Estimating Script

Purpose	To guide the size and time estimating process using the PROBE method
Entry Criteria	- Requirements statement - Size Estimating template and instructions - Size per item data for part types - Time Recording log - Historical size and time data
General	- This script assumes that you are using added and modified size data as the size-accounting types for making size and time estimates. - If you choose some other size-accounting types, replace every "added and modified" in this script with the size-accounting types of your choice.

Step	Activities	Description
1	Conceptual Design	Review the requirements and produce a conceptual design.
2	Parts Additions	Follow the Size Estimating Template instructions to estimate the parts additions and the new reusable parts sizes.
3	Base Parts and Reused Parts	- For the base program, estimate the size of the base, deleted, modified, and added code. - Measure and/or estimate the size of the parts to be reused.
4	Size Estimating Procedure	- If you have sufficient estimated proxy size and actual added and modified size data (three or more points that correlate), use procedure 4A. - If you do not have sufficient estimated data but have sufficient plan added and modified and actual added and modified size data (three or more points that correlate), use procedure 4B. - If you have insufficient data or they do not correlate, use procedure 4C. - If you have no historical data, use procedure 4D.
4A	Size Estimating Procedure 4A	- Using the linear-regression method, calculate the β_0 and β_1 parameters from the estimated proxy size and actual added and modified size data. - If the absolute value of β_0 is not near 0 (less than about 25% of the expected size of the new program), or β_1 is not near 1.0 (between about 0.5 and 2.0), use procedure 4B.
4B	Size Estimating Procedure 4B	- Using the linear-regression method, calculate the β_0 and β_1 parameters from the plan added and modified size and actual added and modified size data. - If the absolute value of β_0 is not near 0 (less than about 25% of the expected size of the new program), or β_1 is not near 1.0 (between about 0.5 and 2.0), use procedure 4C.
4C	Size Estimating Procedure 4C	If you have any data on plan added and modified size and actual added and modified size, set $\beta_0 = 0$ and β_1 = (actual total added and modified size to date/plan total added and modified size to date).
4D	Size Estimating Procedure 4D	If you have no historical data, use your judgment to estimate added and modified size.

(continued)

Continued on next page

PROBE Estimating Script (Continued)

Step	Activities	Description
5	Time Estimating Procedure	- If you have sufficient estimated proxy size and actual development time data (three or more points that correlate), use procedure 5A. - If you do not have sufficient estimated size data but have sufficient plan added and modified size and actual development time data (three or more points that correlate), use procedure 5B. - If you have insufficient data or they do not correlate, use procedure 5C. - If you have no historical data, use procedure 5D.
5A	Time Estimating Procedure 5A	- Using the linear-regression method, calculate the β_0 and β_1 parameters from the estimated proxy size and actual total development time data. - If β_0 is not near 0 (substantially smaller than the expected development time for the new program), or β_1 is not within 50% of 1/(historical productivity), use procedure 5B.
5B	Time Estimating Procedure 5B	- Using the linear-regression method, calculate the β_0 and β_1 regression parameters from the plan added and modified size and actual total development time data. - If β_0 is not near 0 (substantially smaller than the expected development time for the new program), or β_1 is not within 50% of 1/(historical productivity), use procedure 5C.
5C	Time Estimating Procedure 5C	- If you have data on estimated – added and modified size and actual development time, set $\beta_0 = 0$ and β_1 = (actual total development time to date/estimated – total added and modified size to date). - If you have data on plan – added and modified size and actual development time, set $\beta_0 = 0$ and β_1 = (actual total development time to date/plan total added and modified size to date). - If you only have actual time and size data, set $\beta_0 = 0$ and β_1 = (actual total development time to date/actual total added and modified size to date).
5D	Time Estimating Procedure 5D	If you have no historical data, use your judgment to estimate the development time from the estimated added and modified size.
6	Time and Size Prediction Intervals	- If you used regression method A or B, calculate the 70% prediction intervals for the time and size estimates. - If you did not use the regression method or do not know how to calculate the prediction interval, calculate the minimum and maximum development time estimate limits from your historical maximum and minimum productivity for the programs written to date.

Exit Criteria	- Completed estimated and actual entries for all pertinent size categories - Completed PROBE Calculation Worksheet with size and time entries - Plan and actual values entered on the Project Plan Summary

Continued on next page

Development phase

Develop the program following the PSP1.1 development phase script.

PSP1.1 Development Script

Purpose	To guide the development of small programs
Entry Criteria	- Requirements statement - Project Plan Summary form with estimated program size and development time - ***For projects lasting several days or more, completed Task Planning and Schedule Planning templates*** - Time and Defect Recording logs - Defect Type standard and Coding standard

Step	**Activities**	**Description**
1	Design	- Review the requirements and produce a design to meet them. - Record in the Defect Recording log any requirements defects found. - Record time in the Time Recording log.
2	Code	- Implement the design following the Coding standard. - Record in the Defect Recording log any requirements or design defects found. - Record time in the Time Recording log.
3	Compile	- Compile the program until there are no compile errors. - Fix all defects found. - Record defects in the Defect Recording log. - Record time in the Time Recording log.
4	Test	- Test until all tests run without error. - Fix all defects found. - Record defects in the Defect Recording log. - Record time in the Time Recording log. - Complete a Test Report template on the tests conducted and the results obtained.

Exit Criteria	- A thoroughly tested program that conforms to the Coding standard - Completed Test Report template - Completed Time and Defect Recording logs

Verify that you have met all of the exit criteria for the development phase, then proceed to the postmortem phase.

Continued on next page

Postmortem phase

Conduct the postmortem following the PSP1.1 postmortem script.

PSP1.1 Postmortem Script

Purpose	To guide the PSP postmortem process
Entry Criteria	- Problem description and requirements statement - Project Plan Summary form with program size and development time data - ***For projects lasting several days or more, completed Task Planning and Schedule Planning templates*** - Completed Test Report template - Completed Time and Defect Recording logs - A tested and running program that conforms to the coding and size counting standards

Step	**Activities**	**Description**
1	Defect Recording	- Review the Project Plan Summary to verify that all of the defects found in each phase were recorded. - Using your best recollection, record any omitted defects.
2	Defect Data Consistency	- Check that the data on every defect in the Defect Recording log are accurate and complete. - Verify that the numbers of defects injected and removed per phase are reasonable and correct. - Using your best recollection, correct any missing or incorrect defect data.
3	Size	- Count the size of the completed program. - Determine the size of the base, deleted, modified, base additions, reused, new reusable code, and added parts. - Enter these data in the Size Estimating template. - Determine the total program size - Enter this data in the Project Plan Summary form.
4	Time	- Review the completed Time Recording log for errors or omissions. - Using your best recollection, correct any missing or incomplete time data.

Exit Criteria	- A thoroughly tested program that conforms to the coding and size counting standards - Completed Test Report template - Completed Project Plan Summary form - Completed PIP forms describing process problems, improvement suggestions, and lessons learned - Completed Time and Defect Recording logs

Verify that you have met all of the exit criteria for the PSP1.1 postmortem phase, then submit your assignment.

Continued on next page

Submitting your assignment

When you've completed the postmortem phase, submit your assignment package, source code, and test results to the instructor.

The order for the assignment package is
- PSP1.1 Project Plan Summary form
- Test Report template
- PIP form
- Size Estimating template
- PROBE Calculation worksheet
- Time Recording log
- Defect Recording log
- source program listing
- test results

Guidelines and evaluation criteria for program 4

Evaluation criteria

Your process report must be
- complete
- legible
- in the specified order

Your process data must be
- accurate
- precise
- self-consistent

Suggestions

Remember, you should complete this assignment today.

Keep your programs simple. You will learn as much from developing small programs as from large ones.

If you are not sure about something, ask your instructor for clarification.

Software is not a solo business, so you do not have to work alone.
- You must, however, produce your own estimates, designs, code, and completed forms and reports.
- You may have others review your work, and you may change it as a result.
- You should note any help you receive from others in your process report. Log the review time that you and your associates spend, and log the defects found or any changes made.

