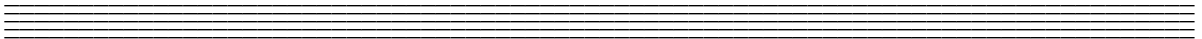


Assignment Kit for Program 2



PSP Fundamentals

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.

This material is approved for public release.
Distribution limited by the Software Engineering Institute to attendees.

PSP Fundamentals

Assignment Kit for Program 2

Overview

Topics

This assignment kit covers the following topics.

Section	See Page
Prerequisites	2
Program 2 requirements	3
Program 2 hints	4
Assignment instructions	5
Guidelines and evaluation criteria	11
PSP1 Grading Checklist	13

Prerequisites

Reading

- Chapters 3, 4, and 5

Assignments

- Coding/Counting standard
-

Program 2 requirements

Program 2 requirements

Using PSP1, write a program to count (in LOC) the

- total program size
- total size of each of the program's parts (classes, functions, or procedures)
- the number of items (or methods) in each part

Produce and print

- a single count for the entire program
- size and item counts for each part together with the part name

If LOC is not a suitable size measure for the language you use, see Chapter 3 and the following “Hints” section for alternative ways to measure size. Then write Program 2 to use those measures.

Use the coding standard defined in your report assignment.

Thoroughly test the program. At a minimum, test the program by counting the total program and part sizes in programs 1 and 2. Example output is illustrated in Table 1. In Table 1, consider the word *part* to be whatever applies for your language: class, function, procedure, and so forth.

Part Name	Number of Items	Part Size	Total Size
ABC	3	86	
DEF	2	8	
GHI	4	92	
			212
...			

Table 1

Program 2 hints

Hints

Do not try to write a sophisticated counting program.

For LOC counting, follow the counting strategy suggested in Chapter 3, page 48, “Size Counters.”

Also consider echoing out the input files with each line pre-pended with a running part count, for example

```
1  this is line 1
2  this is line 2
   this is a comment
3  this is the third line
```

You may hand-in the file with the line numbers for your source programs.

If the classes, functions, embedded code, or anything else in your programming language is hard to identify and count, consider including special comments to identify such things for your counting program. Remember to modify your coding/counting standard to include these comments. You will also have to include such comments in every program, including Program 1.

For database applications or other languages where the LOC measure may not be appropriate, identify the items that you think could provide useful counts. Be as complete as you can while keeping the counting approach simple. A simple sum of the basic countable elements will likely provide as useful a size measure as more sophisticated measures.

Assignment instructions

Assignment instructions

Before starting Program 2, review the top-level PSP1 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 Process Script

Purpose	To guide the development of module-level programs	
Entry Criteria	<ul style="list-style-type: none">- Problem description- PSP1 Project Plan Summary form- <i>Size Estimating template</i>- <i>Historical size and time data (estimated and actual)</i>- Time and Defect Recording logs- Defect Type, Coding, and Size Counting standards- Stopwatch (optional)	
Step	Activities	Description
1	Planning	<ul style="list-style-type: none">- Produce or obtain a requirements statement.- <i>Use the PROBE method to</i> estimate the added and modified size of this program.- <i>Complete the Size Estimating template.</i>- <i>Use the PROBE method to</i> estimate the required development time.- Enter the plan data in the Project Plan Summary form.- Complete the Time Recording log.
2	Development	<ul style="list-style-type: none">- 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	<ul style="list-style-type: none">- A thoroughly tested program- Completed Project Plan Summary form with estimated and actual data- <i>Completed Size Estimating template</i>- <i>Completed Test Report template</i>- Completed PIP forms- Completed Time and Defect Recording logs	

Continued on next page

Assignment instructions, Continued

Planning phase Plan Program 2 following the PSP1 planning phase script.

PSP1 Planning Script

Purpose	To guide the PSP planning process	
Entry Criteria	<ul style="list-style-type: none">- Problem description- PSP1 Project Plan Summary form- <i>Size Estimating template</i>- <i>Historical size and time data (estimated and actual)</i>- Time Recording log	
Step	Activities	Description
1	Program Requirements	<ul style="list-style-type: none">- Produce or obtain a requirements statement for the program.- Ensure that the requirements statement is clear and unambiguous.- Resolve any questions.
2	Size Estimate	<ul style="list-style-type: none">- <i>Produce a program conceptual design.</i>- <i>Use the PROBE method to</i> estimate the added and modified size of this program.- <i>Complete the Size Estimating template and Project Plan Summary form.</i>
3	Resource Estimate	<ul style="list-style-type: none">- <i>Use the PROBE method to</i> 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.)
Exit Criteria	<ul style="list-style-type: none">- Documented requirements statement- <i>Program conceptual design</i>- <i>Completed Size Estimating template</i>- 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, and **then have an instructor review your plan**. After your plan has been reviewed, proceed to the development phase.

Continued on next page

Assignment instructions, Continued

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	<ul style="list-style-type: none"> - Requirements statement - Size Estimating template and instructions - Size per item data for part types - Time Recording log - Historical size and time data
General	<ul style="list-style-type: none"> - 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	<ul style="list-style-type: none"> - 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	<ul style="list-style-type: none"> - 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	<ul style="list-style-type: none"> - 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	<ul style="list-style-type: none"> - 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 $\beta_1 = (\text{actual total added and modified size to date} / \text{plan total added and modified size to date})$.
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

Assignment instructions, Continued

PROBE Estimating Script (Continued)

Step	Activities	Description
5	Time Estimating Procedure	<ul style="list-style-type: none"> - 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	<ul style="list-style-type: none"> - 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	<ul style="list-style-type: none"> - 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	<ul style="list-style-type: none"> - If you have data on estimated – added and modified size and actual development time, set $\beta_0 = 0$ and $\beta_1 = (\text{actual total development time to date}/\text{estimated – total added and modified size to date})$. - If you have data on plan – added and modified size and actual development time, set $\beta_0 = 0$ and $\beta_1 = (\text{actual total development time to date}/\text{plan total added and modified size to date})$. - If you only have actual time and size data, set $\beta_0 = 0$ and $\beta_1 = (\text{actual total development time to date}/\text{actual total added and modified size to date})$.
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	<ul style="list-style-type: none"> - 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		<ul style="list-style-type: none"> - 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

Assignment instructions, Continued

Development phase

Develop the program following the PSP1 development phase script.

PSP1 Development Script

Purpose	To guide the development of small programs	
Entry Criteria	<ul style="list-style-type: none">- Requirements statement- Project Plan Summary form with estimated program size and development time- Time and Defect Recording logs- Defect Type standard and Coding standard	
Step	Activities	Description
1	Design	<ul style="list-style-type: none">- 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	<ul style="list-style-type: none">- 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	<ul style="list-style-type: none">- 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	<ul style="list-style-type: none">- 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.- <i>Complete a Test Report template on the tests conducted and the results obtained.</i>
Exit Criteria	<ul style="list-style-type: none">- A thoroughly tested program that conforms to the Coding standard- <i>Completed Test Report template</i>- Completed Time and Defect Recording logs	

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

Continued on next page

Assignment instructions, Continued

Postmortem phase

Conduct the postmortem following the PSP1 postmortem script.

PSP1 Postmortem Script

Purpose	To guide the PSP postmortem process	
Entry Criteria	<ul style="list-style-type: none">- Problem description and requirements statement- Project Plan Summary form with program size and development time data- <i>Completed Test Report template</i>- 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	<ul style="list-style-type: none">- 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	<ul style="list-style-type: none">- 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	<ul style="list-style-type: none">- Count the size of the completed program.- <i>Determine the size of the base, deleted, modified, base additions, reused, new reusable code, and added parts.</i>- <i>Enter these data in the Size Estimating template.</i>- <i>Determine the total program size</i>- <i>Enter this data in the Project Plan Summary form.</i>
4	Time	<ul style="list-style-type: none">- Review the completed Time Recording log for errors or omissions.- Using your best recollection, correct any missing or incomplete time data.
Exit Criteria	<ul style="list-style-type: none">- A thoroughly tested program that conforms to the coding and size counting standards- <i>Completed Test Report template</i>- 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 postmortem phase, and then review your assignment.

Continued on next page

Guidelines and evaluation criteria for Program 2

Reviewing your assignment

Use the attached grading checklist to check your assignment. Ensure that your assignment is correct before you submit it.

Your process data must be

- complete
 - accurate
 - precise
 - self-consistent
-

Submitting your assignment

When you've completed your review, package the following data files into a zip file and upload the zip file to the program 2 assignment page on the SEI Learning Portal.

- Process data (mdb export file from SEI Student Workbook or zip data backup file from Process Dashboard).
 - Source program listings for programs 1 and 2.
 - Test results.
 - Test report doc file (Process Dashboard only).
 - PIP form doc file (Process Dashboard only).
 - Coding/Counting standard report.
-

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.
-

Grading Checklist - PSP1

Student _____
Instructor _____

Program _____

Accepted or Resubmit	Comments
Accepted	
Resubmit	

Legend	√ - O.K.	X - resubmit	sw - SEI Student Workbook	pd - Process Dashboard
---------------	----------	--------------	----------------------------------	-------------------------------

Assignment Package	Comments
All files are included	
Process data file { *.mdb (sw) or *.zip (pd) }	
Source program listings for programs 1 and 2	
Test results	
Test report .doc file (pd only)	
PIP form .doc file (pd only)	
Coding/Counting standard report	

Program and Test Results	Comments
The program appears to be workable.	
All required tests have been run.	
The actual output is correct for each test.	
Source is compatible with Coding/Counting standard.	

Test Report Template	Comments
The test report is complete.	
Planned and actual results are included for all required tests.	
All information to repeat the tests is provided.	

Time Log	Comments
Time data are entered for all process steps.	
Process steps are sequenced appropriately.	
Time data are entered against the appropriate process step.	
Interrupt time is tracked appropriately.	
Time data are complete and reasonable.	
Times were recorded as the work was done.	

Defect Log	Comments
Every defect has all required data.	
Every defect, injection phase precedes removal phase.	
Every defect has a fix time.	
Defects injected in compile and test have fix numbers.	
Defect descriptions describe what was changed.	

Grading Checklist - PSP1

<input type="checkbox"/>	Defect types are consistent with description.	
<input type="checkbox"/>	Defect types are consistent with phase injected.	
<input type="checkbox"/>	Defect types are assigned consistently.	

Size Estimating Template & PROBE Worksheet

Comments

<input type="checkbox"/>	<i>Plan and actual size data are complete and reasonable.</i>	
<input type="checkbox"/>	<i>The reuse and base measures are used correctly.</i>	
<input type="checkbox"/>	<i>A suitable number of new parts are identified.</i>	
<input type="checkbox"/>	<i>The item sizes are balanced around medium.</i>	
<input type="checkbox"/>	<i>The relative size data values are correct and based on historical data.</i>	
<input type="checkbox"/>	<i>The appropriate PROBE method for size has been selected.</i>	
<input type="checkbox"/>	<i>The appropriate PROBE method for effort has been selected</i>	

Planning Summary

Comments

<input type="checkbox"/>	Actual size data are entered correctly.	
<input type="checkbox"/>	<i>Planned and actual size/hour data are reasonable.</i>	

PIP Form

Comments

<input type="checkbox"/>	<i>The PIP form is completed.</i>	
<input type="checkbox"/>	<i>The entries show insight and thought.</i>	

Consistency Checks

Comments

<input type="checkbox"/>	Defects removed are consistent with compile and test phase time and program size.	
<input type="checkbox"/>	Total compile defect fix times are close to and no greater than compile time.	
<input type="checkbox"/>	Total test defect fix times are close to and no greater than test time.	
<input type="checkbox"/>	Defect dates & phases are consistent with the time log.	
<input type="checkbox"/>	<i>Actual Added on planning summary close to and no less than actual BA+PA on size estimating template.</i>	

General

Comments

<input type="checkbox"/>	Followed the defined process.	
<input type="checkbox"/>	Complete, consistent, and accurate process data was collected.	
<input type="checkbox"/>	The student did his or her own work.	
<input type="checkbox"/>	<i>Historical data are used in planning the work.</i>	