

OS Problem Set Comments

Introduction

This document describes the expected outcomes for the OS Problem Set Assignment. You must include your name and a description within each deliverable. The written work is as important as the coding sections and must be handled appropriately. In the following sections of this document, a discussion on the expectations of each deliverable is presented. Many of the sections may contain more detail than was expected in your assignment. However, the added detail is appropriate for this type of assignment and should be used as a guideline for the future.

Design

The design portion of the assignment must provide a detailed discussion of the components used in the system. Information contained in the design document should have been explicitly detailed. Once again, this document must contain sufficient detail such that a programmer can implement your design but does not include the actual code that resulted from your design. It is necessary to include a title page and a table of contents with your design.

The following sections outline the subject matter that should have been included in your design document.

Scope/Overview

The scope/overview should have introduced the problem and the design. Information contained in this section should give an overview of what is to follow, as well as stating any limitations.

Discuss the approach that was used to implement a simulation of the SIMMAC machine/OS problem. In this deliverable it is necessary to use prose not just simple narratives. Your work should establish any assumptions for the assignment and explain your direction. In this section you should have discussed the following:

- Introduction and overview
- The design of the study (not the actual design of the simulator, but your approach to a solution)
- Assumptions
- Criteria establishment - How should it work?
- Criteria validation - How do you know it is valid?
- Procedures
- Materials and/or special apparatus
- Plans for data analysis

In essence, this section would have covered the steps taken to produce the resulting design document. This portion of the assignment provides the background on the solution.

Data Design

In this section, the design should have discussed each of the data items that are used in the system. A definition and discussion on each of the following items should have been provided in this section:

- Input data
- SIMMAC program loading
- Process control block information
- Process Queue
- Output data

The detail should have included the purpose for each item, the size, and any other information necessary for the item's implementation.

Architectural Design

This section will discuss, in detail, each of the system components. The design should have included the initialization, scheduler, and termination components. There must be a text description of the main modules of the system, and a diagram showing the relationships between these modules in your system. Each module's responsibilities should be clearly delineated as well as showing the information or data flows between the modules. If there are timing relationships between modules (IE. "the initialization module runs before"), they should be documented.

Initialization

Initialization for this assignment must include the details of the following issues:

- Startup
- Machine initialization
- Data input
- Input data organization
- Overall SIMMAC bootstrap
- SIMMAC structure

Task Manager

A task manager for this assignment must include the details of the following issues:

- Initialization

- Parsing of the input
- Initial job selection
- Context Switch
- Job rescheduling
- PCB data management
- Termination

Termination

- Data formatting
- Data output
- Shutdown

Class and Object Design/Modules

This section provides the details of the classes and objects used in the modules that were designed for the system. It is not acceptable to extract portions of the actual code and include them in this section. The details include:

- Class Names
- Method Names
- Modifiers
- Arguments
- Error Messages
- Files accessed
- Files Changed
- Methods called
- Narratives
- Pseudo code
- Diagrams

Interface Design

You may have included any human-computer interaction, external interfaces, or internal interfaces in another section. If not, this section should have discussed the following:

- Any attended operation
- Access to logs
- Access to files
- Other

Test Provisions

This section would include your testing and verification designs.

Program Simulation/Output

The zip file should have emptied its contents into one subdirectory. From there “javac *.java” must compile correctly, then “java app . . . “ should run the program. Modifying paths or bat file is not acceptable.

The following is a list of factors that were used to evaluate the simulation:

- Input
- Organization
- SIMMAC design/implementation
- Context switch algorithm
- Job selection algorithm implementation
- Output of the given input distribution
- Output of test distributions