



Date: 05/09/2017

CPU_Scheduling_Simulation_Project

CPU scheduling is a process which allows one process to use the CPU while the execution of another process is on hold (in waiting state) due to unavailability of any resource like I/O etc, thereby making full use of CPU. The aim of CPU scheduling is to make the system efficient, fast and fair.

Team Information : Team Number 5

Team Members

Name	Email	Github Username
Akkaraboina, Manju Yadav	manju1404@gmail.com	infinitepassion
Ashoknaidu Gedela	gedela.huf031@gmail.com	ashoknaidug
Biradhavolu, Sai Avinash Reddy	avinash9240@gmail.com	avinash9240

Code that is attributed by each team member.

Integrated all different snippets from team members. we have simulationa.py, simulationb.py, simulationc.py in driver. and we have a Package component, scheduling, processes, io display and load file are handled in them separately

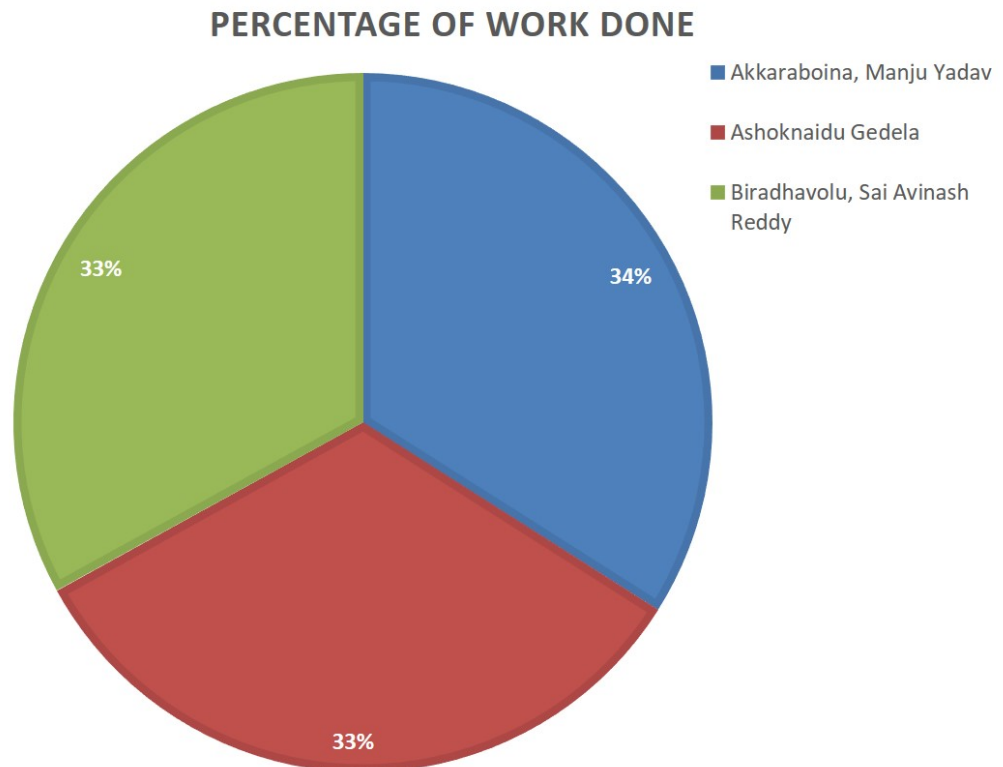
- Akkaraboina, Manju Yadav
 - Implemented all the concepts for process.
 - Implemented Process I/O s.
 - Implemented Semaphores to facilitate and restrict access to shared resources.
 - Implemented Queues for Semaphores and functions.
 - Integrated all different snippets from team members to make code work as required.
 - Verified to match required outputs.
- Ashoknaidu Gedela
 - Implemented all the concepts for process.
 - Implemented Job Scheduling Queue and Ready Queue.
 - Implemented Memory Queue, ppu queues for size of memory of each Process.

- Implemented Semaphores to facilitate and restrict access to shared resources.
- Integrated all different snippets from team members to make code work as required.
- Verified to match required outputs.
- Biradhavolu, Sai Avinash Reddy
 - Implemented all the concepts for process.
 - Implemented I/O wait Queue.
 - Implemented I/O Process to work as required (RUNNING, READY, or BLOCKED..etc).
 - Implemented I/O Status Queue and shared I/O wait Queue.
 - Implemented Semaphores to facilitate and restrict access to shared resources.
 - Integrated all different snippets from team members to make code work as required.
 - Verified to match required outputs.

Time each team member spent working on project.

Soon after Spring break, every team member put together their efforts to complete project on time. It almost took 10 days together to integrate all snippets and to make sure it works as desired.

Visualization of Participation in Pie Chart.



Readme.md



Files in project.

-We have three file names simulationa.py, simulationb.py, simulationc.py separately to run for the three input files. -We have a package file "componets" where each file has different modules of the simulation -The files in package are loadfiles, event

files and scheduling files -Our simulation file calls the scheduling.py file to perform all the simulation activities. -This file acts as a center to the all the programs. -We read the entire input data into dictionaries and then read the data dictionary and schedule our events. -We have a directory input_data, which has all the input source files and the output files.

