**Contents Page**

1. **J2SE Certificate by Feasible Technology**
2. **Company Profile with web Name**
3. **Java Programming Language**

3a. Introduction

3b. Feature of Java

3c. First Java Program (Hello World Program)

3d. Topics Covered

1. **Project Work (CPU Scheduling Algorithm Implementation**)

4a. Introduction

4b. CPU scheduling algorithms

4c. CPU scheduling software

4d. about this software

4e. How to use cpu scheduling software

1. **Snapshot of the Project**
2. **Summary**

6a. Aim

6b. Test

6c. Conclusion

**Java Programming Language**

**Introduction**

Java is a general-purpose computer-programming language that is concurrent, class-based, object-oriented, and specifically designed to have as few implementation dependencies as possible. It is intended to let application developers "write once, run anywhere" (WORA), meaning that compiled Java code can run on all platforms that support Java without the need for recompilation. Java applications are typically compiled to byte code that can run on any Java virtual machine (JVM) regardless of computer architecture.

**Feature of Java** **Programming Language**

**Object-oriented**

Java is object-oriented programming language. Everything in Java is an object. Object-oriented means we organize our software as a combination of different types of objects that incorporates both data and behavior.

Object-oriented programming (OOPs) is a methodology that simplifies software development and maintenance by providing some rules.

Basic concepts of OOPs are:

1. Object

2. Class

3. Inheritance

4. Polymorphism

5. Abstraction

6. Encapsulation

## ****Platform Independence****

Java code is compiled into intermediate format (byte code), which can be executed on any systems for which Java virtual machine is ported. That means you can write a Java program once and run it on Windows, Mac, Linux or Solaris without re-compiling. Thus the slogan “*Write once, run anywhere*” of Java.

## ****Robust****

With automatic garbage collection and simple memory management model (no pointers like C/C++), plus language features like generics, try-with-resources, Java guides programmer toward reliable programming habits for creating highly reliable applications.

## ****Secure****

The Java platform is designed with security features built into the language and runtime system such as static type-checking at compile time and runtime checking (security manager), which let you creating applications that can’t be invaded from outside. You never hear about viruses attacking Java applications.

**High Performance**

Java code is compiled into byte code which is highly optimized by the Java compiler, so that the Java virtual machine (JVM) can execute Java applications at full speed. In addition, compute-intensive code can be re-written in native code and interfaced with Java platform via *Java Native Interface* (JNI) thus improve the performance.

**Multi Threading**

Java multithreading feature makes it possible to write program that can do many tasks simultaneously. Benefit of multithreading is that it utilizes same memory and other resources to execute multiple threads at the same time, like While typing, grammatical errors are checked along.

**Portable**

Java is portable because it facilitates you to carry the java byte code to any

Platform. It does not require any type of implementation.

**Topic Covered**

a. Basic Programming

b. Jdk, Jre and Jvm

c. Object and class

d. Access modifier

e. Inheritance

f. Abstract class

g. Interface

h. Packages

i. Exception handling

j. Netbeans (Project Development Demonstration)

k. Data Base Connectivity

**Hello World Program**

*class HelloWorld*

*{*

*public static void main (String[]args)*

*{*

*System.out.println('HELLOWORLD");  
}  
}*

**About Program**

* class keyword is used to declare a class in java.
* public keyword is an access modifier which represents visibility, it means it is visible to all.
* static is a keyword, if we declare any method as static, advantage of static method is that there is no need to create object to invoke the static method. The main method is executed by the JVM, so it doesn't require to create object to invoke the main method. So it saves memory.
* void is the return type of the method, it means it doesn't return any value.
* main represents the starting point of the program.
* String[] args is used for command line argument.
* System.out.println () is used print statement.

**Project Work (CPU Scheduling Algorithm Implementation**)

**Introduction**

**“*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.***

***Whenever the CPU becomes idle, the operating system must select one of the processes in the ready queue to be executed. The selection process is carried out by the short-term scheduler (or CPU scheduler). The scheduler selects from among the processes in memory that are ready to execute, and allocates the CPU to one of them.”***

**CPU SCHDULING TYPES**

1. First come First Serve
2. Shortest Job First (sjf) non pre- emptive
3. Shortest Remaining Time (srt) pre –emptive
4. Priority Scheduling
5. Round Robin Scheduling

**First come First Serve**

* Jobs are executed on first come, first serve basis.
* It is a non-preemptive, pre-emptive scheduling algorithm.
* Easy to understand and implement.
* Its implementation is based on FIFO queue.
* Poor in performance as average wait time is high.

**Shortest Job First**

* This is also known as **shortest job first**, or SJF
* This is a non-preemptive, pre-emptive scheduling algorithm.
* Best approach to minimize waiting time.
* Easy to implement in Batch systems where required CPU time is known in advance.
* Impossible to implement in interactive systems where required CPU time is not known.
* The processer should know in advance how much time process will take.

**Shortest Remaining Time (Pre-emptive SJF )**

* Shortest remaining time (SRT) is the preemptive version of the SJN algorithm.
* The processor is allocated to the job closest to completion but it can be preempted by a newer ready job with shorter time to completion.
* Impossible to implement in interactive systems where required CPU time is not known.
* It is often used in batch environments where short jobs need to give preference.

**Priority Based CPU Scheduling (**preemptive**)**

* Priority scheduling is a non-preemptive and preemptive algorithm and one of the most common scheduling algorithms in batch systems.
* Each process is assigned a priority. Process with highest priority is to be executed first and so on.
* Processes with same priority are executed on first come first served basis.
* Priority can be decided based on memory requirements, time requirements or any other resource requirement.

**Round Robin Scheduling**

* Round Robin is the preemptive process scheduling algorithm.
* Each process is provided a fix time to execute, it is called a **quantum**.
* Once a process is executed for a given time period, it is preempted and other process executes for a given time period.
* Context switching is used to save states of preempted processes.
* This maintain a queue , after execution till given time quantum it place itself in back of the queue

**CPU Scheduling Software**

**Platform-** Netbeans

**Language –** Java **(**core java**)**

**Data required –** No. of the Process, Process id, Arrival time, Burst time, Priority of the process, Time Quantum.

**Application –** Calculate completion time, turnaround time, waiting time, avg. waiting time and avg. turnaround time with Graphical Gantt chart.

**About this software**

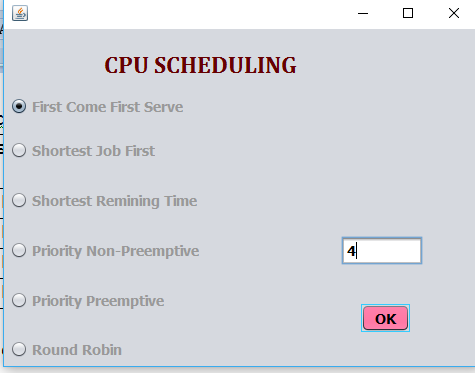
* This software has 7 frames, 6 for the different algorithms and 1 is the welcome page.
* Welcome page has 6 radioButtons which are for selecting an algorithm, One label for the title of the software and a textField to enter number of the process, One button is also there for go to selected algorithm’s frame.
* Algorithm frame consist one table, one button, two textField and three labels.
* Table content vary from different algorithm, there requirement and working method.
* textField is for result.(avg. waiting and turnaround time)

**How to use the CPU scheduling Software**

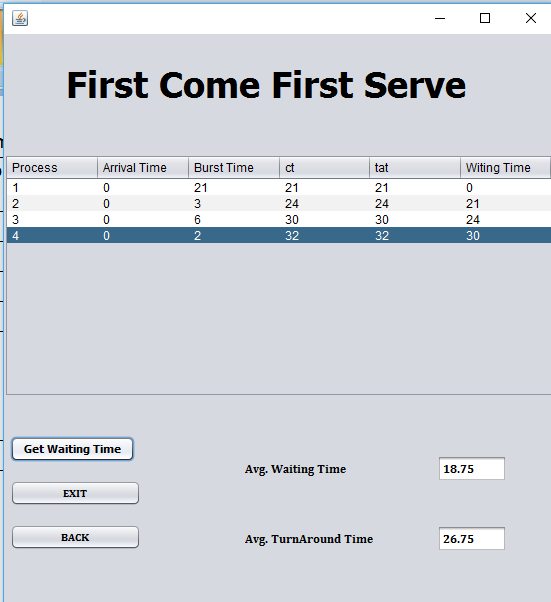
* When you run the software a welcome page is opened initially. You just need to select a desired algorithm and number of the process is entered in the box given.
* Click the “Ok” button
* Selected Algorithm frame is opened now, for the further calculation you need to provide some details like process id, arrival time and burst time is common for the entire algorithm. Some algorithms need priority of the processes and the time quantum.
* After filling all the required information, you are ready to get result.
* Click the “get result” button. Result will be display as soon as you entered the button.
* When you do it manually it takes at least 15 to 20 min. to solve a problem and find waiting time for those processes. It depends upon your accuracy and practice.
* Here this Software is well written and tested that I can surely say that this will give you 99% time correct answer.

**SNAPSHOT**

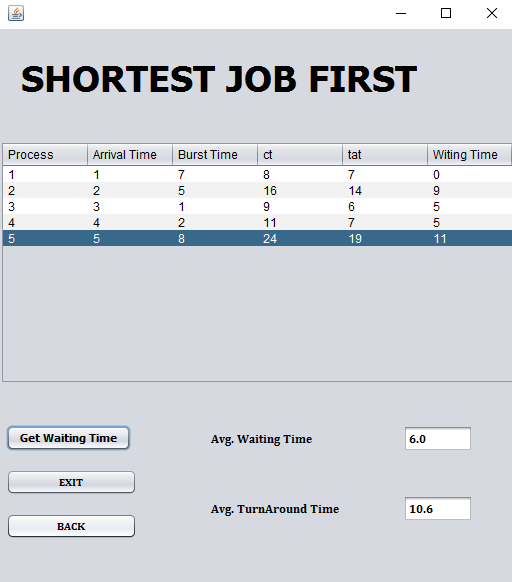
* **WELCOME PAGE**

****

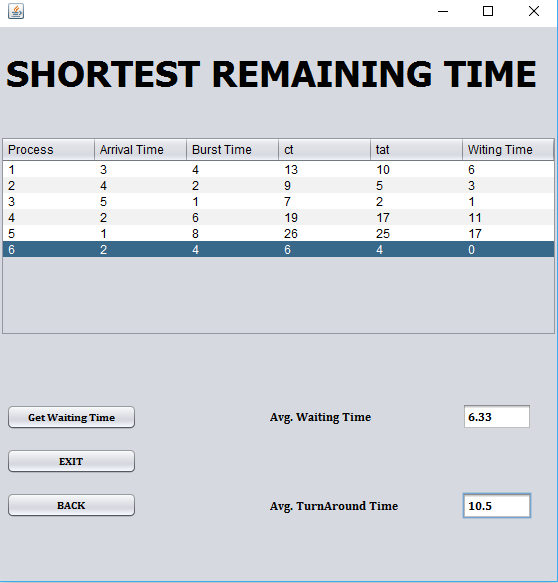
* **FIRST COME FIRST SERVE**

****

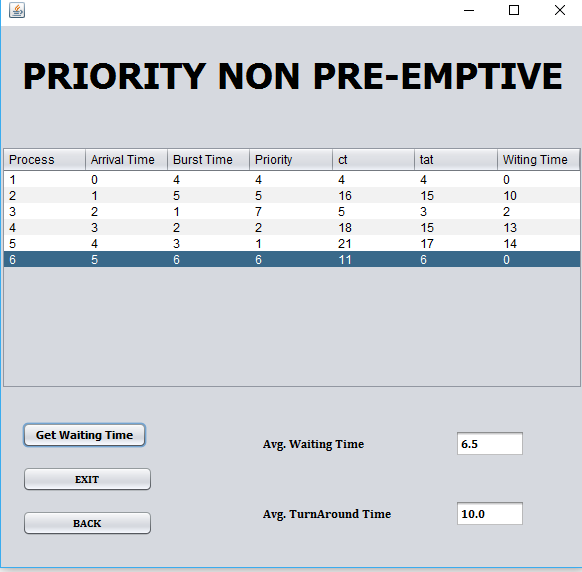
* **SHORTEST JOB FIRST**

****

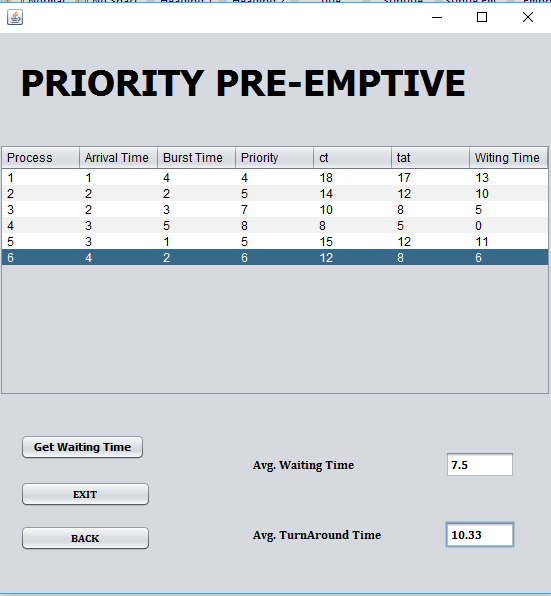
* **SHORTEST REMAINING TIME**

****

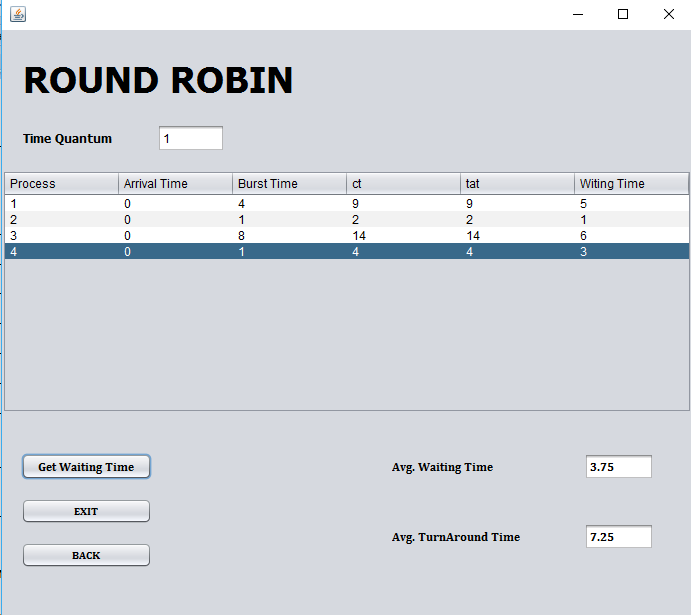
* **PRIORITY NON-PREEMPTIVE**

****

**PRIORITY PREEMPTIVE**

****

* **ROUND ROBIN**



**SUMMARY**

* **AIM**

To implement the theoretical knowledge to the working model, as we have studied CPU scheduling in 4th sem subject named operating system. In summer training 2018 I have learned java programming language. By this project it is my attempt to implement those training knowledge to the practical one.

* **TEST**

I have tested this software for almost all the condition and possible inputs. Still there are possibilities of error or exception. I am glad to solve them.

* **CONCLUSION**
  1. I enjoyed making my first project.
  2. I really work hard while making this project and learn something new each day.
  3. I have used core java concepts, Netbeans platform to make this project.
  4. Summer training is very good, feasible technology helps me to enhance my knowledge and development of the software.
  5. I would like to continue this spirit and make useful software which makes life easy.