ASSIGNMENT C4

TITLE : Process scheduling algorithms in Android and Tizen
Problem Statement:
Study assignment on process scheduling algorithms in Android and Tizen

Objectives:

- To understand Android OS
- To understand Tizen OS
- To understand Concept of process management

Outcomes:
I will be have

- Knowledge of Android and tizen OS
- Study of process management in android and tizen OS.
- Application of android and tizen os

Software and Hardware Requirements:

- Working PC.
- 64 bit Fedora OS
- Eclipse IDE and JAVA
- I3 processor

Theory

Android OS : - Android is a mobile operating system developed by Google, based on a modified version of the Linux kernel and other open source software and designed primarily for touchscreen mobile devices such as smartphones and tablets. In addition, Google has further developed Android TV for televisions, Android Auto for cars, and Android Wear for wrist watches, each with a specialized user interface. Variants of Android are also used on game consoles, digital cameras, PCs and other electronics.

- Initially developed by Android Inc., which Google bought in 2005, Android was unveiled in 2007, with the first commercial Android device launched in September 2008. The operating system has since gone through multiple major releases, with the current version being 8.1 "Oreo", released in December 2017.

- The android is a powerful operating system and it supports a large number of applications in Smartphones. These applications are more comfortable and advanced for the users. The hardware that supports android software based on ARM architecture platform. The android is an open source operating system means that it's free and any one can use it. The android has got millions of apps available that can help you managing your life one or other way and it is available low cost in market at that reasons android is very popular.

- The android development supports the full java programming language. Even other packages that are API and JSE are not supported. The first version 1.0 of android development kit (SDK) was released in 2008 and the latest updated version is jelly bean.

Some android versions :

- Gingerbread (2.3)
- Honeycomb (3.0)
- Ice Cream Sandwich (4.0)
- Jelly Bean (4.3/4.2/4.1)
- KitKat (4.4)
- Lollipop (5.0)
- Marshmallow (6.0)
- Nougat (7.0)
- Oreo (8.0)

- **Tizen** is a mobile operating system developed by Samsung that runs on a wide range of Samsung devices, including smartphones; tablets; in-vehicle infotainment (IVI) devices; smart televisions; smart cameras; smartwatches; Blu-ray players; smart home appliances (refrigerators, lighting, washing machines, air conditioners, ovens/microwaves); and robotic vacuum cleaners.

- In 2010 Samsung was developing the Samsung Linux Platform (SLP) for the LiMo Foundation, whilst Intel and Nokia were leading the MeeGo project, another open source Linux mobile OS. In 2011 the MeeGo project was abandoned by its peers with Intel joining forces with Samsung to create Tizen, a new project based on code from SLP. The Linux Foundation also cancelled support of MeeGo in favor of Tizen. In 2013 Samsung merged its homegrown Bada project into Tizen.

- The Tizen Association was formed to guide the industry role of Tizen, including requirements gathering, identifying and facilitating service models, and overall industry marketing and education.Members of the Tizen Association represent major sectors of the mobility industry. Current members include: Fujitsu, Huawei, Intel, KT, NEC Casio, NTT DoCoMo, Orange, Panasonic, Samsung, SK Telecom, Sprint and Vodafone

- Samsung announced in November 2016 that they would be collaborating with Microsoft to bring .Net support to Tizen.

- Samsung is currently the only Tizen member developing and using the operating system.

- As of 2017 Tizen is second largest smartwatch platform, behind watchOS and ahead of Android Wear

- On January 1, 2012, the LiMo Foundation was renamed Tizen Association. The Tizen Association works closely with the Linux Foundation, which supports the Tizen open source project.

- April 30, 2012: Tizen 1.0 released.
- February 18, 2013: Tizen 2.0 released.
- May 20, 2017: Tizen 3.0 released

- The first Tizen tablet was shipped by Systena in October 2013. Part of a development kit exclusive to Japan, it was a 10-inch quad-core ARM with 1920× 1200 resolution

PROCESS SCHEDULING ALGORITHMS IN ANDROID AND TIZEN OS :

Normal scheduling

Android is based on Linux and uses the Linux kernel's scheduling mechanisms for determining scheduling policies. This is also true for Java code and threads.

The Linux's time slice scheduling policy combines static and dynamic priorities. Processes can be given an

initial priority from 19 to -20 (very low to very high priority). This priority will assure that higher priority processes will get more CPU time when needed. These level are however dynamic, low level priority tasks that do not consume their CPU time will fine their dynamic priority increased. This dynamic behaviour results is an overall better responsiveness.

In terms of dynamic priorities it is ensured that lower priority processes will always have a lower dynamic priority than processes with real-time priorities.

Android uses two different mechanisms when scheduling the Linux kernel to perform process level scheduling .

Real-time scheduling

The standard Linux kernel provides two real-time scheduling policies, SCHED_FIFO and SCHED_RR. The main real-time policy is SCHED_FIFO. It implements a first-in, first-out scheduling algorithm. When a SCHED_FIFO task starts running, it continues to run until it voluntarily yields the processor, blocks or is preempted by a higher-priority real-time task. It has no timeslices. All other tasks of lower priority will not be scheduled until it relinquishes the CPU. Two equal-priority SCHED_FIFO tasks do not preempt each other. SCHED_RR is similar to SCHED_FIFO, except that such tasks are allotted timeslices based on their priority and run until they exhaust their timeslice. Non-real-time tasks use the SCHED_NORMAL scheduling policy (older kernels had a policy named SCHED_OTHER). **Thread Scheduling**

A thread scheduler decides which threads in the Android system should run, when, and for how long Android's thread scheduler uses two main factors to determine the scheduling:

- Niceness Values
- Control Groups (Cgroups)

Niceness Values

A thread with a higher niceness value will run less often than those with a lower niceness value (this sounds paradoxical)

it is possible to change the priority via:

- o *thread.setPriority(int priority)* - values: 0 (least prioritized) to 10 (most prioritized)
- o *process.setThreadPriority(int priority)* - values: -20 (most prioritized) to 19 (least prioritized)

Control Groups (Cgroups)

- Android has multiple control groups. The most important are:

- the Foreground Group
- the Background Group

• every thread belongs to a thread control group (e.g. Foreground Group)

• threads in the different control groups are allocated different amounts of CPU execution time

• threads in the Foreground Group receive a lot more execution time than threads in the Background Group

Priority Based Pre-Emptive Task Scheduling for Android Operating System

The key concept present in any operating system which allows the system to support multitasking, multiprocessing, etc. is Task Scheduling . Task Scheduling is the core which refers to the way the different processes are allowed to share the common CPU. Scheduler and dispatcher are the softwares which help to carry out this assignment . Android operating system uses O (1) scheduling algorithm as it is based on Linux Kernel 2.6. Therefore the scheduler is names as Completely Fair Scheduler as the processes can schedule within a constant amount of time, regardless of how many processes are running on the operating system . Preemptive task scheduling involves interrupting the low priority tasks when high priority tasks are present in the queue. This scheduling is particularly used for mobile operating system as the CPU utilization is medium, turnaround time and response time is high. Mobile phones are required to meet specific time deadlines for the tasks to occur.

Fixed-priority preemptive scheduling is a scheduling system commonly used in real-time systems. With fixed priority preemptive scheduling, the scheduler ensures that at any given time, the processor executes the highest priority task of all those tasks that are currently ready to execute. The preemptive scheduler has a clock interrupt task that can provide the scheduler with options to switch after the task has had a given period to execute—the time slice. This scheduling system has the advantage of making sure no task hogs the processor for any time longer than the time slice. However, this scheduling scheme is vulnerable to process or thread lockout: since priority is given to higher-priority tasks, the lower-priority tasks could wait an indefinite amount of time. One common method of arbitrating this situation is aging, which gradually increments the priority of waiting processes and threads, ensuring that they will all eventually execute.

Dynamic priority pre-emptive scheduling

Earliest-deadline first scheduling: a job's priority is inversely proportional to its absolute deadline. The difference between deadline monotonic scheduling and earliest-deadline first scheduling is that DM is a static priority algorithm, EDF is a dynamic priority algorithm.]EDF can guarantee that all deadlines are met.

Conclusion:

Thus , I have studied concept of process scheduling of Android and Tizen Operating System.