UCS6611 - MOBILE APPLICATION DEVELOPMENT LABORATORY

Ex. No. 5 implement an application that uses multi-threading

Date: 9/9/21 **Name:** Srinath S

Class: CSE-C **Roll:** 185001205

Question:

implement an application that uses multi-threading

Code:

ActivityMain.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

        <Button
            android:id="@+id/button"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginBottom="500dp"
            android:text="Start"
            app:layout_constraintBottom_toBottomOf="parent"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintHorizontal_bias="0.500"
            app:layout_constraintStart_toStartOf="parent" />

        <ProgressBar
            android:id="@+id/progressBar"
            style="?android:attr/progressBarStyleHorizontal"
            android:layout_width="300dp"
            android:layout_height="33dp"
            app:layout_constraintBottom_toTopOf="@+id/button"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintHorizontal_bias="0.496"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toTopOf="parent"
            app:layout_constraintVertical_bias="0.783" />

        <TextView
            android:id="@+id/textView"
```

```
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Initiate Task"
        app:layout_constraintBottom_toTopOf="@+id/progressBar"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.72" />

    <View
        android:id="@+id/divider"
        android:layout_width="409dp"
        android:layout_height="1dp"
        android:background="?android:attr/listDivider"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.431"
        tools:layout_editor_absoluteX="1dp" />

    <TextView
        android:id="@+id/textView2"
        android:layout_width="180dp"
        android:layout_height="35dp"
        android:layout_marginTop="68dp"
        android:layout_marginEnd="24dp"
        android:text="Enter Sleep Time"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toStartOf="@+id/input"
        app:layout_constraintTop_toBottomOf="@+id/divider"
        app:layout_constraintVertical_bias="0.0" />

    <EditText
        android:id="@+id/input"
        android:layout_width="164dp"
        android:layout_height="46dp"
        android:layout_marginEnd="16dp"
        android:layout_marginBottom="300dp"
        android:ems="10"
        android:inputType="textPersonName"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent" />

    <Button
        android:id="@+id/button3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="184dp"
        android:text="Sleep"
```

```
            app:layout_constraintBottom_toBottomOf="parent"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintHorizontal_bias="0.498"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toBottomOf="@+id/divider"
            app:layout_constraintVertical_bias="0.098" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

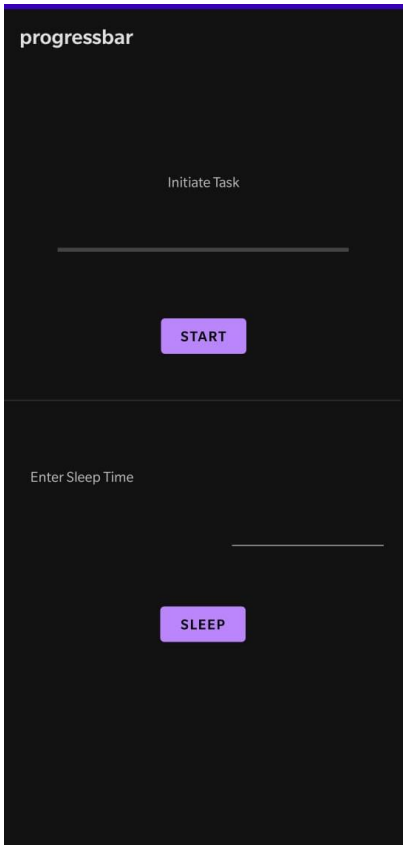
Mainactivity.java:

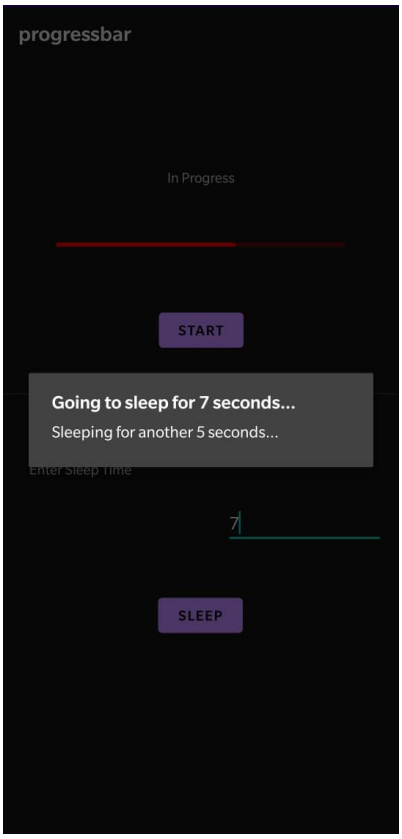
```
package com.example.progressbar;

import androidx.appcompat.app.AlertDialog;
import androidx.appcompat.app.AppCompatActivity;

import android.graphics.Color;
import android.os.Bundle;
import android.os.CountDownTimer;
import android.os.Handler;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ProgressBar;
import android.widget.TextView;

public class MainActivity extends AppCompatActivity {
    int progress=0;
    ProgressBar progressBar;
    Handler hand = new Handler();
    Button start,reset,sleep;
    TextView value;
    EditText input;
    AlertDialog.Builder builder;
    AlertDialog alertDialog;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        progressBar = findViewById(R.id.progressBar);
        start = findViewById(R.id.button);
        sleep = findViewById(R.id.button3);
        value = findViewById(R.id.textView);
        input = findViewById(R.id.input);
        progressBar.setProgress(progress);
        builder=new AlertDialog.Builder(this);
```

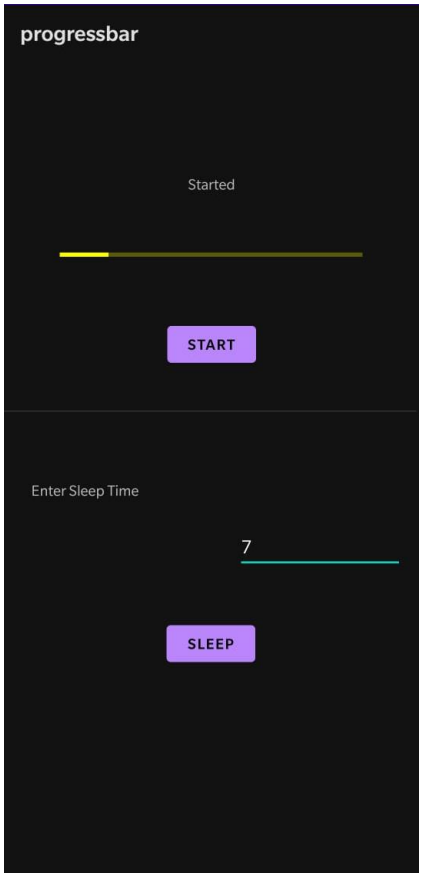
```
        start.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                runnable.run();
            }
        });

        sleep.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                final int sleepTime =
Integer.parseInt(input.getText().toString());
                System.out.println(sleepTime);
                builder.setTitle("Going to sleep for "+sleepTime+"
seconds...").setMessage("Going to sleep for "+sleepTime+" seconds...");
                alertDialog=builder.create();
                alertDialog.show();

                new Thread(new Runnable() {
                    @Override
                    public void run() {
                        try {
                            Thread.sleep(sleepTime*1000);
                        }
                        catch (Exception e) {
                            e.printStackTrace();
                        }
                        finally {
                            alertDialog.dismiss();
                        }
                    }
                }).start();

                new CountDownTimer(sleepTime*1000, 1000) {
                    public void onTick(long millisUntilFinished) {
                        alertDialog.setMessage("Sleeping for another
"+(millisUntilFinished/1000)+" seconds...");
                    }

                    public void onFinish() {
                        alertDialog.setMessage("Done!");
                    }
                }.start();
            }
        });
    }

    Runnable runnable = new Runnable() {
```

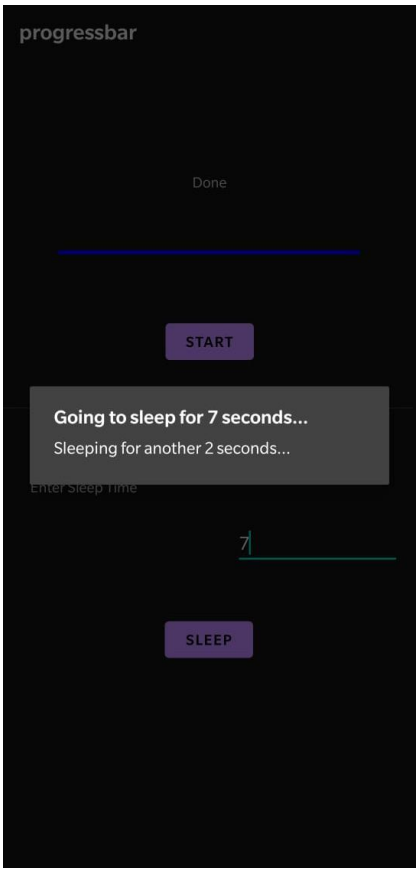
```
        @Override
        public void run() {
            try {
                doTask();
            }
            catch (Exception e) {
                e.printStackTrace();
            }
            finally {
                hand.postDelayed(runnable, 100);
            }
        }
    };

    private void doTask() {
        progress+=2;
        if(progress<=25){
            value.setText("Started");
            progressBar.getProgressDrawable().setColorFilter(Color.YELLOW,
android.graphics.PorterDuff.Mode.SRC_IN);
        }
        else if(progress<=80){
            value.setText("In Progress");
            progressBar.getProgressDrawable().setColorFilter(Color.RED,
android.graphics.PorterDuff.Mode.SRC_IN);
        }
        else if(progress<=90){
            value.setText("In Progress");
            progressBar.getProgressDrawable().setColorFilter(Color.BLUE,
android.graphics.PorterDuff.Mode.SRC_IN);
        }
        else if(progress==100){
            value.setText("Done");
        }

        if(progress>100) {
            hand.removeCallbacks(runnable);
        }
        else {
            progressBar.setProgress(progress);
        }
    }
}
```

Output:


progressbar
Initiate Task
START
Enter Sleep Time
SLEEP


progressbar
In Progress
START
Going to sleep for 7 seconds...
Sleeping for another 5 seconds...
7
SLEEP


progressbar
Started
START
Enter Sleep Time
7
SLEEP


progressbar
Done
START
Going to sleep for 7 seconds...
Sleeping for another 2 seconds...
7
SLEEP