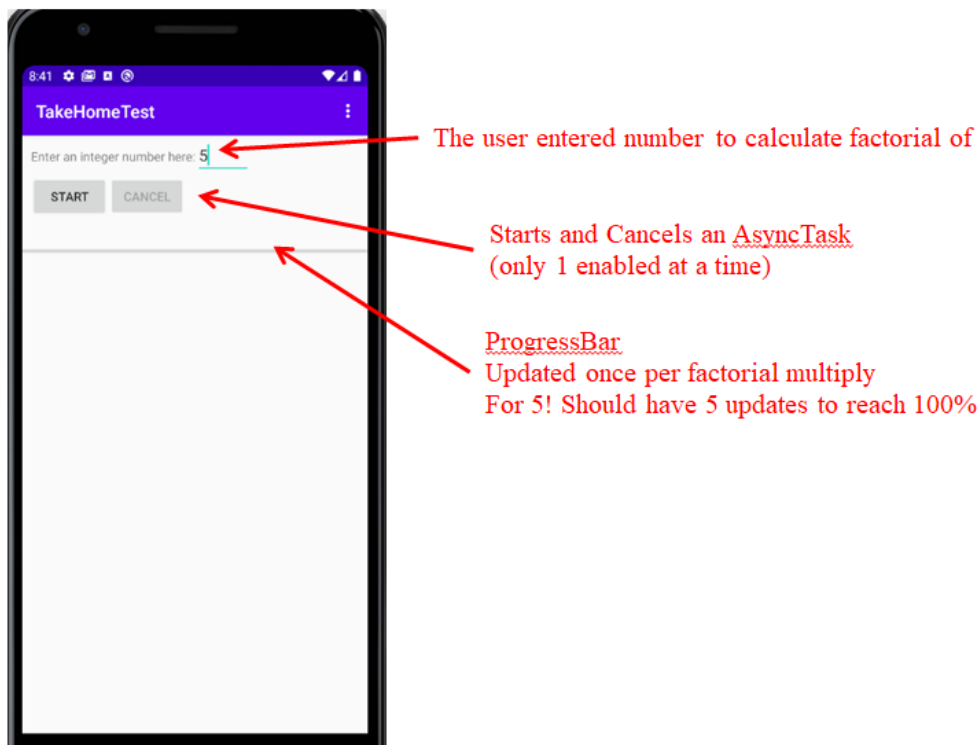


## 475/575

This test is an individual effort. You may use any resources except for other people.  
Gradle requirements; compileSdkVersion=29, minSdkVersion=16, targetSdkVersion=29  
I will compile and test on a Google Pixel running API 29  
Please use sensible defaults, and write high quality code. Please run appropriate error checks as needed. Do not worry about device configuration changes.

**You are to write a threaded factorial calculator (ie  $3! = 3*2*1$ ).  
The MainActivity should look like the following;**

**30 pts**



1. When Start pressed
  - a. Get integer number (**val**) from EditText to calculate factorial of ( for above its 5)
  - b. Verify `val > 10` (if not pop a toast indicating it must be `> 10` and exit)
  - c. Disable Start button, Enable Cancel button
  - d. Get **delay** from defaultsharedpreferences (see SettingsActivity below)
  - e. Create and start an async task to do the factorial. At a minimum, pass it **val** and **delay**
2. When Cancel pressed
  - a. Cancel the AsyncTask

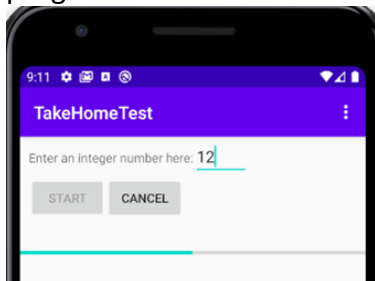
The AsyncTask- does the factorial, is cancelable, updates the Start and Cancel buttons, and updates the ProgressBar in the MainActivity

30 pts

doInBackground here is my code with a couple of TODOs

```
protected String doInBackground(Integer... integers) {  
    //number to apply factorial operation to  
    int fact_val = integers[0];  
  
    //will hold the final calculated factorial value  
    result=1;  
  
    for (int i =1; i<=fact_val;i++) {  
        //TODO handle cancel  
  
        //sleep for delay milliseconds  
        try {  
            Thread.sleep(delay);  
        } catch (InterruptedException e) {  
            e.printStackTrace();  
        }  
  
        //1 round of the factorial  
        result = result * i;  
  
        //TODO update MainActivity ProgressBar  
    }  
    return SUCCESS;  
}
```

**Note** this is how progress looks in the MainActivity when 6 of the 11 multiplications have occurred; these updates are a direct result of the AsyncTask updating the main thread with its progress.



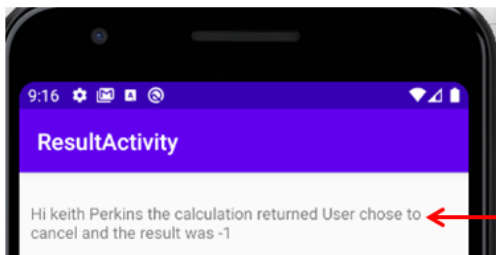
If the user cancels the AsyncTask via the Cancel button the Async task should return “User Chose to Cancel” and result should= -1

If the AsyncTask is NOT canceled, the AsyncTask should return “Success” and result = factorial calculation.

Once finished, the AsyncTasks should;

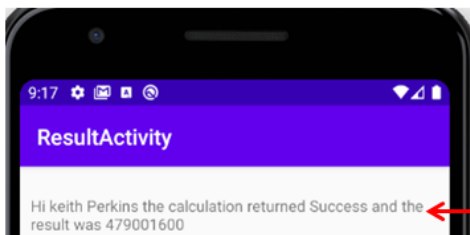
- Enable Start button, Disable Cancel button
  - place the above string (either “User Chose to Cancel” or “Success”) and the factorial result in a bundle
  - create ResultActivity and passing the above bundle to it
  - start ResultActivity
- 

**The ResultActivity should look like the following; it just builds a string to display**  
**15pts**



Canceled output

Get the name from defaultSharedPreferences  
Get resultstring (User Chose To cancel) from bundle  
Get -1 from bundle



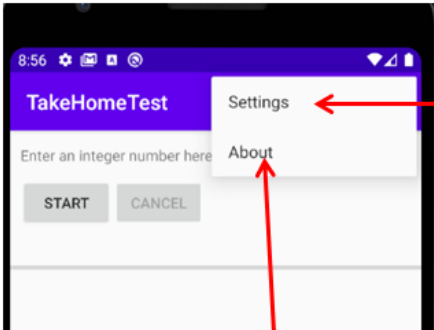
Not Canceled output

Get the name from defaultSharedPreferences  
Get resultstring (Success) from bundle  
Get 479001600 from bundle

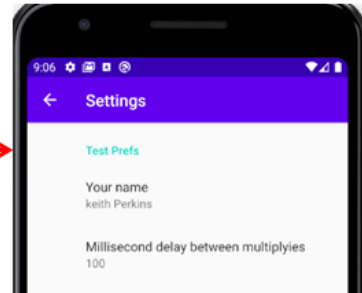
## The SettingsActivity and AboutDialog

20 pts settings activity

10 pts Dialog



Select Settings,  
instantiates  
SettingsActivity  
You must be  
able to enter a  
name and a  
delay time.



Select 'About' pulls name from  
defaultsharedpreferences  
(set with above right SettingsActivity)  
Then creates this dialog

