## ĐỀ 1:

### Câu 1

**Với đầu vào là tên của bạn, hãy cho biết chữ cái nhỏ nhất không xuất hiện trong chuỗi tên của bạn (giả sử tất cả các chữ cái không viết hoa) là gì? Viết mã giả chương trình Java thực hiện điều này. Ví dụ: Tên Nguyen Van A => nguyenvana . Chương trình in ra chữ b**

Nhớ THAY Nguyen Van A bằng tên của mình

|  |
| --- |
| import java.util.HashSet;  public class SmallestMissingLetter {  public static void main(String[] args) {  String name = "Nguyen Van A"; // Tên của bạn  // Chuyển đổi chuỗi tên thành chữ thường và loại bỏ khoảng trắng  String lowercaseName = name.toLowerCase().replaceAll(" ", "");  // Tạo một HashSet để lưu trữ các chữ cái xuất hiện trong chuỗi tên  HashSet<Character> letters = new HashSet<>();  // Duyệt qua từng ký tự trong chuỗi tên  for (char c : lowercaseName.toCharArray()) {  // Nếu ký tự là một chữ cái, thêm nó vào HashSet  if (Character.isLetter(c)) {  letters.add(c);  }  }  // Tìm chữ cái nhỏ nhất không xuất hiện trong chuỗi tên  char smallestMissingLetter = 'a';  while (letters.contains(smallestMissingLetter)) {  smallestMissingLetter++;  }  System.out.println("Chữ cái nhỏ nhất không xuất hiện trong chuỗi tên là: " + smallestMissingLetter);  }  } |

### 

### Câu 2

**Android là gì, activity là gì? cho ví dụ minh họa? Liệt kê một vài ưu điểm của hệ thống Android**

-Android là Hệ điều hành tối ưu cho di động, dựa trên nhân linux, dòng vi xử lý ARM - Có thể được tùy biến cho thiết bị di động và những hệ thống nhúng

-Android được phát triển và hỗ trợ bởi liên minh OHA (Open Handset Allien) gồm nhiều công ty phần cứng, phần mềm và dịch vụ: Google, HTC, LG, Samsung, Motorola, Sprint, T-Mobile, NVIDIA, Intel, Broadcom, Qualcom,… Có 2 phiên bản song song: Android & Google API

VD: Android 1.1. Android 1.5, Android 1.6

<https://didongviet.vn/dchannel/he-dieu-hanh-android/>

Một activity là một màn hình giao diện, một ứng dụng gồm một hoặc nhiều activity Android OS cung cấp sẵn một số lượng khá lớn các activity tiêu chuẩn Ví dụ: Giao diện quay số và gọi điện Giao diện settings Lập trình viên có thể tự viết activity riêng hoặc sử dụng các activity đã có

ƯU ĐIỂM của hệ thống Android:

Đa luồng (multithread) Web ready (html5, css3, javascript, flash) Open GL Java Đa chạm (multitouch) Media (full HD video, mpeg4, H.264, mp3,…) Network ready (wifi, 3g, bluetooth,…) GPS Sensors

* Kho ứng dụng đa dạng: Hiện nay, trên CH Play đã có tổng cộng trên 3 triệu ứng dụng khác nhau. Mọi ứng dụng phục vụ công việc, học tập và giải trí đều có thể được tìm thấy ở trên đây.
* Mẫu mã đa dạng: Có rất nhiều hãng sản xuất điện thoại đều sử dụng hệ điều hành Android cho thiết bị của hãng. Điều đó giúp mang đến nhiều mẫu điện thoại hấp dẫn khác nhau với đa dạng phân khúc giá, giúp mọi người dùng dễ dàng tiếp cận với thiết bị Android hơn.
* Có thể mở rộng bộ nhớ bằng thẻ nhớ: Bên cạnh bộ nhớ trong của máy, bạn có thể mở rộng bộ nhớ bằng việc gắn thêm thẻ SD dung lượng cao.
* Khả năng tùy biến cao: Hệ điều hành Android hoạt động với bản chất mã nguồn mở. Do đó bất kỳ ai cũng có thể lấy mã nguồn của hệ điều hành này. Vì vậy các nhà sản xuất và các lập trình viên có thể tùy biến chỉnh sửa để mang đến phiên bản tốt nhất dành cho điện thoại thông minh của hãng.
* Được nhiều người dùng ưa chuộng: Giao diện của Android được đánh giá là dễ dàng sử dụng, không tốn nhiều thời gian để làm quen. Đồng thời nếu như gặp bất kỳ vấn đề nào trong khi sử dụng, bạn cũng sẽ được hỗ trợ vô cùng nhiệt tình từ cộng đồng Android.

### Câu 3

**Sự khác biệt giữa margin và padding trong android?**

Trong Android, margin và padding là hai thuộc tính quan trọng để điều chỉnh khoảng cách giữa các phần tử trong giao diện người dùng. Dưới đây là sự khác biệt giữa margin và padding:

1. Margin: Margin là khoảng cách bên ngoài của một phần tử so với phần tử khác xung quanh nó. Nó tạo ra khoảng trống giữa phần tử và các phần tử xung quanh. Margin không gian này không bị ảnh hưởng bởi nội dung hoặc các thuộc tính khác của phần tử. Nó chỉ định khoảng cách từ biên ngoài của một phần tử đến biên ngoài của phần tử kế tiếp hoặc phần tử cha.
2. Padding: Padding là khoảng trống bên trong của một phần tử, nằm giữa nội dung của phần tử và biên của nó. Nó tạo ra khoảng trống xung quanh nội dung của phần tử, giữa nội dung và biên của phần tử. Padding ảnh hưởng đến không gian xung quanh nội dung của phần tử.

Margin là khoảng trống bên ngoài phần tử

Padding là khoảng trống bên trong phần tử

### Câu 4

**Tệp tin AndroidManifest có chức năng gì trong Android? Giải thích các thành phần cơ bản trong tệp tin này?**

“AndroidManifest.xml” chứa các khai báo về ứng dụng Qua việc phân tích nội dung của file, hệ thống biết ứng dụng có thể dùng vào việc gì

Các thông tin cơ bản trong “AndroidManifest.xml”:

* Các thông tin về ứng dụng (tên package, tên ứng dụng, biểu tượng của ứng dụng,…)
* Các quyền cần có để chạy ứng dụng (quyền truy xuất internet, quyền đọc contact, quyền đọc SD card,…)
* Phiên bản API tối thiểu có thể chạy ứng dụng
* Các tính năng phần cứng cần thiết cho ứng dụng (GPS, camera, bluetooth,…)
* Các bộ API liên kết sử dụng trong ứng dụng (Google Maps API, AdMod,…)
* Cấu hình màn hình khởi chạy (ngang/dọc,
* Mô tả về các activity (màn hình) của ứng dụng, Thông tin về activity (tên activity, tên class,….). Xác định xem activity nào là giao diện khởi động của ứng dụng
* Mô tả về các service (dịch vụ) mà ứng dụng cung cấp, Thông tin về service (tên dịch vụ, class xử lý dịch vụ,…)
* Mô tả về các broadcast receiver mà ứng dụng cung cấp •
* Thông tin về receiver (tên receiver, class xử lý,…), Các loại tín hiệu gửi đến receiver
* Mô tả về các content provider mà ứng dụng cung cấp, Các đối tượng truy xuất content provider, Các quyền truy xuất content provider

Cụ thể:

1. <manifest>: Thẻ gốc của tệp tin AndroidManifest.xml. Nó chứa thông tin chung về ứng dụng và các thành phần của nó.
2. <application>: Thẻ này định nghĩa thông tin về ứng dụng và chứa các thành phần của ứng dụng như Activities, Services, Receivers và Providers.
3. <activity>: Thẻ này định nghĩa một hoạt động (Activity) trong ứng dụng. Activity đại diện cho một màn hình giao diện người dùng và quản lý tương tác với người dùng.
4. <service>: Thẻ này định nghĩa một dịch vụ (Service) trong ứng dụng. Service là một thành phần chạy ngầm, không có giao diện người dùng, được sử dụng để thực hiện các tác vụ dài hạn hoặc không liên quan trực tiếp đến giao diện người dùng.
5. <receiver>: Thẻ này định nghĩa một bộ thu (Receiver) trong ứng dụng. Receiver là một thành phần chịu trách nhiệm nhận và xử lý các thông báo (Intents) từ hệ thống hoặc các ứng dụng khác.
6. <provider>: Thẻ này định nghĩa một cung cấp (Content Provider) trong ứng dụng. Content Provider là một thành phần cho phép ứng dụng chia sẻ dữ liệu với các ứng dụng khác thông qua giao diện thống nhất.

### Câu 5

**Phương thức nào dùng để ánh xạ đến các view của Android thông qua thuộc tính id? Giải thích tại sao?**

**findViewById(int id)**

Trong Android, để ánh xạ đến các view thông qua thuộc tính id, ta sử dụng phương thức findViewById().

Phương thức findViewById() được sử dụng để tìm và lấy tham chiếu (reference) đến một view cụ thể trong cấu trúc giao diện người dùng của ứng dụng dựa trên giá trị của thuộc tính id.

Tham số truyền vào cho phương thức findViewById() là kiểu int để chỉ định id của view mà bạn muốn tìm kiếm. Mã thông thường sẽ truyền vào id dưới dạng int thông qua tệp tài nguyên R.java của ứng dụng.

Ví dụ, nếu bạn có một Button trong XML với id là "myButton", bạn sẽ truyền id này vào phương thức findViewById() dưới dạng int như sau: findViewById(R.id.myButton).

### Câu 6

**Thành phần nào dùng để truyền dữ liệu giữa các activities trong Android trong số các thành phần sau đây? Nêu những hiểu biết của bạn về thành phần này?**

Intent

Intent là một đối tượng được sử dụng để giao tiếp và truyền dữ liệu giữa các thành phần khác nhau trong ứng dụng Android, bao gồm giữa các Activities. Nó có thể chứa thông tin như dữ liệu cần truyền, hành động muốn thực hiện và thông tin về thành phần đích mà Intent sẽ được gửi đến.

Intent có thể được sử dụng với các phương thức như startActivity() để chuyển từ một Activity hiện tại sang một Activity mới, hoặc startActivityForResult() để chuyển đến một Activity và nhận kết quả trả về từ nó.

Intent có thể chứa dữ liệu bằng cách sử dụng các extras (cặp key-value) để truyền thông tin. Dữ liệu có thể là các kiểu dữ liệu cơ bản như số nguyên, chuỗi, boolean, hoặc cả các đối tượng tùy chỉnh. Khi nhận được Intent, Activity đích có thể trích xuất dữ liệu từ extras để sử dụng trong quá trình xử lý.

Ngoài việc truyền dữ liệu, Intent cũng cho phép gửi các tín hiệu hành động (action) như ACTION\_VIEW để mở một URI, ACTION\_SEND để chia sẻ nội dung và nhiều tín hiệu khác.

### Câu 7

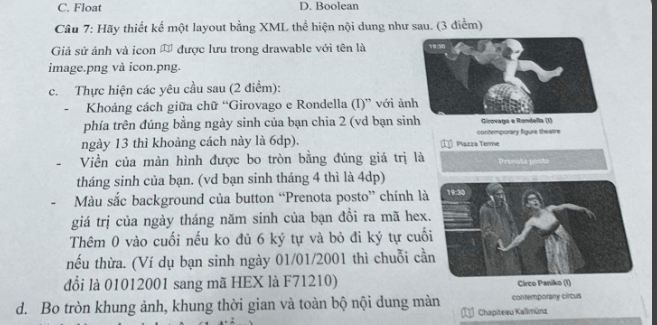
**Hãy thiết kế layout XML thể hiện nội dung như sau:**

**Gỉa sử ảnh và icon được lưu trong drawable với tên là image.png và icon.png**

1. **Thực hiện các yêu cầu:**

* **Khoảng cách giữa chữ “geogre” và ảnh phía trên đúng bằng ngày sinh của bạn chia 2 (9th February 1999). Ví dụ sinh ngày 9 thì khoảng cách là 4.5dp**
* **Viền của màn hình được bo tròn bằng đúng giá trị là tháng sinh của bạn (vd bạn sinh tháng 2 thì giá trị là 2 dp)**
* **Màu sắc background của button “posto” là giá trị ngày tháng năm sinh của bạn đổi ra mã hex. Thêm 0 vào cuối nếu ko đủ 6 kí tự, hoặc bỏ đi kí tự cuối nếu thừa.**

1. **bo tròn khung ảnh, khung thời gian và toàn bộ nội dung màn hình đúng như ảnh trên.**

****

|  |
| --- |
| **<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"**  **android:layout\_width="match\_parent"**  **android:layout\_height="match\_parent"**  **android:background="@drawable/background"**  **android:padding="@dimen/month\_padding">**  **<ImageView**  **android:id="@+id/profile\_image"**  **android:layout\_width="150dp"**  **android:layout\_height="150dp"**  **android:layout\_centerHorizontal="true"**  **android:layout\_marginTop="@dimen/name\_image\_margin"**  **android:src="@drawable/image"**  **android:background="@drawable/image\_border"**  **android:scaleType="centerCrop" />**  **<TextView**  **android:id="@+id/name\_text"**  **android:layout\_width="wrap\_content"**  **android:layout\_height="wrap\_content"**  **android:layout\_below="@id/profile\_image"**  **android:layout\_centerHorizontal="true"**  **android:layout\_marginTop="@dimen/name\_image\_margin"**  **android:text="geogre"**  **android:textColor="@color/text\_color"**  **android:textSize="16sp" />**  **<TextView**  **android:id="@+id/birth\_date\_text"**  **android:layout\_width="wrap\_content"**  **android:layout\_height="wrap\_content"**  **android:layout\_below="@id/name\_text"**  **android:layout\_centerHorizontal="true"**  **android:layout\_marginTop="@dimen/birth\_date\_margin"**  **android:text="9th February 1999"**  **android:textColor="@color/text\_color"**  **android:textSize="14sp" />**  **<Button**  **android:id="@+id/post\_button"**  **android:layout\_width="wrap\_content"**  **android:layout\_height="wrap\_content"**  **android:layout\_below="@id/birth\_date\_text"**  **android:layout\_centerHorizontal="true"**  **android:layout\_marginTop="@dimen/post\_button\_margin"**  **android:text="posto"**  **android:textColor="#000000"**  **android:background="@drawable/post\_button\_background" />**  **</RelativeLayout>** |

* **@dimen/month\_padding là giá trị tương ứng với kích thước bo tròn viền màn hình (tháng sinh của bạn).**
* **@dimen/name\_image\_margin là giá trị tương ứng với khoảng cách giữa chữ "geogre" và ảnh phía trên (ngày sinh của bạn chia 2).**
* **@dimen/birth\_date\_margin là giá trị tương ứng với khoảng cách giữa ngày sinh và chữ "posto" phía dưới.**
* **@dimen/post\_button\_margin là giá trị tương ứng với khoảng cách giữa chữ "posto" và viền dưới cùng của màn hình.**

**@drawable/background, @drawable/image\_border và @drawable/post\_button\_background là các tệp tin drawable tương ứng với hình nền, khung ảnh và nút "posto" mà bạn có thể tạo trong thư mục res/drawable của dự án.**

**@color/text\_color là mã màu tương ứng với màu chữ của các thành phần.**

**Lưu ý rằng bạn cần thay đổi giá trị dimen và drawable**

## ĐỀ 2:

### Câu 1:

**Với đầu vào là 1 dãy các chữ số 0 và 1. Hãy viết mã giả chương trình Java đưa ra số chữ số cần phải xóa nhỏ nhất để dãy luôn là dãy số không giảm. (số 0 luôn đứng trước số 1). Ví dụ 0010 thì xóa số 0 ở cuối là 001 => in ra 1. Hoặc 011001 -> xóa 2 số 1 ở vị trí cạnh số 0 đầu tiên còn 0001 => in ra 2.**

|  |
| --- |
| public class DecreasingDigits {  public static int countMinDeletions(String input) {  int deletions = 0;  int countOnes = 0;  for (int i = 0; i < input.length(); i++) {  if (input.charAt(i) == '0') {  if (countOnes > 0) {  deletions += countOnes;  countOnes = 0;  }  } else {  countOnes++;  }  }  deletions += countOnes;  return deletions;  }  public static void main(String[] args) {  String input = "011001";  int minDeletions = countMinDeletions(input);  System.out.println("Number of deletions needed: " + minDeletions);  }  } |

### Câu 2

**Android là gì, activity là gì? cho ví dụ minh họa? Trình bày vòng đời của 1 activity trong hệ thống Android**

-Android là Hệ điều hành tối ưu cho di động, dựa trên nhân linux, dòng vi xử lý ARM - Có thể được tùy biến cho thiết bị di động và những hệ thống nhúng

-Android được phát triển và hỗ trợ bởi liên minh OHA (Open Handset Allien) gồm nhiều công ty phần cứng, phần mềm và dịch vụ: Google, HTC, LG, Samsung, Motorola, Sprint, T-Mobile, NVIDIA, Intel, Broadcom, Qualcom,… Có 2 phiên bản song song: Android & Google API

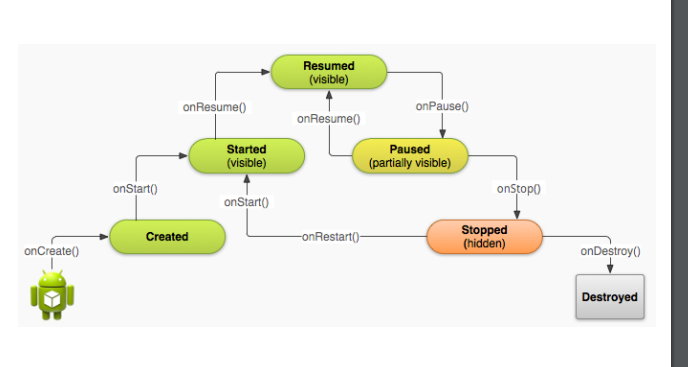
VD: Android 1.1. Android 1.5, Android 1.6

<https://didongviet.vn/dchannel/he-dieu-hanh-android/>

Một activity là một màn hình giao diện, một ứng dụng gồm một hoặc nhiều activity Android OS cung cấp sẵn một số lượng khá lớn các activity tiêu chuẩn Ví dụ: Giao diện quay số và gọi điện Giao diện settings Lập trình viên có thể tự viết activity riêng hoặc sử dụng các activity đã có

Vòng đời của một activity

OnCreate => OnStart => OnResume => OnPause => OnStop => OnReStart => OnDestroy



Một activity có bốn trạng thái:

* Active hay Running: activity đang chạy trên màn hình
* Paused: khi một activity mất focus nhưng vẫn đang chạy trên màn hình (một activity trong suốt hoặc một activity không chiếm toàn bộ màn hình thiết bị đè lên)
* Stopped: khi một activity bị che khuất hoàn toàn bởi một activity khác
* Killed hay Shutdown: khi một activity đang Paused hay Stopped, hệ thống có thể xóa activity ấy nếu cần (chẳng hạn như cần bộ nhớ vào việc khác)

Các hàm trong vòng đời

onCreate(...): gọi khi activity khởi tạo

onStart(): gọi khi acivity xuất hiện trên màn hình

onResume(): gọi ngay sau onStart hoặc người dùng focus, hàm này đưa ứng dụng lên top màn hình

onPause(): gọi khi hệ thống focus đến activity khác

onStop(): gọi khi activity bị che hoàn toàn

onRestart(): gọi khi ứng dụng khởi chạy lại

onDestroy(): gọi khi ứng dụng chuẩn bị được gỡ khỏi bộ nhớ

### Câu 3

**Để mở máy ảo Android Emulator ta chọn mục nào? Nghĩa của cụm từ viết tắt là gì?**

AVD Manager => Android Virtual Device Manager (Tiếng Anh dễ hiểu mà)

### Câu 4

**Thành phần nào dùng để truyền dữ liệu giữa các activities trong Android trong số các thành phần sau đây? Nêu những hiểu biết của bạn về thành phần này?**

Intent

Intent là một đối tượng được sử dụng để giao tiếp và truyền dữ liệu giữa các thành phần khác nhau trong ứng dụng Android, bao gồm giữa các Activities. Nó có thể chứa thông tin như dữ liệu cần truyền, hành động muốn thực hiện và thông tin về thành phần đích mà Intent sẽ được gửi đến.

Intent có thể được sử dụng với các phương thức như startActivity() để chuyển từ một Activity hiện tại sang một Activity mới, hoặc startActivityForResult() để chuyển đến một Activity và nhận kết quả trả về từ nó.

Intent có thể chứa dữ liệu bằng cách sử dụng các extras (cặp key-value) để truyền thông tin. Dữ liệu có thể là các kiểu dữ liệu cơ bản như số nguyên, chuỗi, boolean, hoặc cả các đối tượng tùy chỉnh. Khi nhận được Intent, Activity đích có thể trích xuất dữ liệu từ extras để sử dụng trong quá trình xử lý.

Ngoài việc truyền dữ liệu, Intent cũng cho phép gửi các tín hiệu hành động (action) như ACTION\_VIEW để mở một URI, ACTION\_SEND để chia sẻ nội dung và nhiều tín hiệu khác.

### Câu 5

**Listview là gì? Nêu cách thức khai báo và sử dụng 1 Listview trong file xml và trong java. Sử dụng Adapter trong Listview để làm gì. CHo ví dụ minh họa.**

ListView là một thành phần giao diện người dùng trong Android để hiển thị danh sách dữ liệu có thể cuộn. Nó cho phép bạn hiển thị một danh sách các mục dạng dọc và người dùng có thể cuộn lên xuống để xem các mục khác.

Để khai báo và sử dụng một ListView trong file XML, bạn có thể sử dụng mã sau:

|  |
| --- |
| <ListView  android:id="@+id/listView"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  /> |

Trong mã Java, bạn cần thực hiện các bước sau để sử dụng ListView:

|  |
| --- |
| // tạo listview  ListView listView = findViewById(R.id.listView);  //Chuẩn bị dữ liệu cho ListView, thường là một mảng hoặc danh sách dữ liệu  String[] data = {"Item 1", "Item 2", "Item 3"};  //Tạo một Adapter để gắn kết dữ liệu với ListView:  ArrayAdapter<String> adapter = new ArrayAdapter<>(this, android.R.layout.simple\_list\_item\_1, data);  //Gán Adapter cho ListView:  listView.setAdapter(adapter); |

Adapter trong ListView được sử dụng để kết nối dữ liệu và giao diện người dùng.

Nó giúp ListView hiểu cách hiển thị dữ liệu và tạo các view cho mỗi mục trong danh sách.

Adapter cung cấp phương thức để tạo view cho mỗi mục và liên kết dữ liệu tương ứng.

Ví dụ minh họa: Giả sử bạn có một danh sách tên người dùng và bạn muốn hiển thị danh sách này trong một ListView. Dưới đây là ví dụ cách sử dụng ListView và Adapter để làm điều đó:

|  |
| --- |
| ListView listView = findViewById(R.id.listView);  String[] usernames = {"John", "Jane", "Mike", "Emily"};  ArrayAdapter<String> adapter = new ArrayAdapter<>(this, android.R.layout.simple\_list\_item\_1, usernames);  listView.setAdapter(adapter); |

### Câu 6

Thuộc tính Android:layout\_alignParentTop nhận giá trị kiểu gì? Thuộc tính này dùng để làm gì và được dùng trong layout nào?

Nhận giá trị kiểu Boolean : true or false

Thuộc tính này được sử dụng để xác định vị trí của một phần tử giao diện người dùng (View) trong quan hệ với phần tử cha của nó. Khi thuộc tính android:layout\_alignParentTop được đặt là true cho một View, nó sẽ được căn chỉnh với phía trên của phần tử cha của nó.

Thuộc tính android:layout\_alignParentTop thường được sử dụng trong các loại layout như RelativeLayout, trong đó bạn muốn đặt một View cố định ở phía trên của phần tử cha. Ví dụ:

### Câu 7 tương tự đề 1