

Nome: _____

Assinatura: _____

Número: _____

Esta prova é constituída por 50 questões divididas por várias secções. Cada questão vale 0,2 valores e a classificação da componente teórica desta cadeira é obtida da soma das classificações destas perguntas (um total de 10 valores).

Pressupõe-se que as respostas a esta prova são dadas no próprio enunciado. Por isso, faça por não a rasurar, sob pena de ver respostas anuladas. Leia o enunciado das perguntas atentamente. O enunciado indica se a resposta é composta por uma

só opção, ou por uma ou mais. No caso das respostas serem compostas por mais do que uma opção, só é considerada certa a resposta que tem todas as opções certas, a não ser que expresse em contrário. Normalmente, uma das opções desconta metade da cotação atribuída a essa pergunta. Caso a resposta tenha apenas duas opções, a escolha da errada penaliza sempre. Quaisquer tentativas de fraude ou utilização de dispositivos eletrónicos durante a resolução desta prova são penalizadas com a sua anulação.

Definição de Dispositivo Móvel e Desenvolvimento de Interfaces de Utilizador

Q1.: Das opções seguintes, selecione aquelas que correspondem a características que os dispositivos devem apresentar para serem considerados dispositivos móveis. Os dispositivos móveis devem:

- ☒ Ter, obrigatoriamente, capacidade de processamento;
- ☒ Poder ser operados com uma ou duas mãos;
- ☒ Ter, obrigatoriamente, tecnologia de comunicação com outros dispositivos móveis;
- ☒ Ter, obrigatoriamente, um peso que um humano com idade compreendida entre os 18 e os 37 anos possa suportar, de modo a que possa ser transportado;
- ☒ Ter, obrigatoriamente, um dispositivo de entrada / saída, para interação com o utilizador.

Q2.: Nas interfaces de utilizador orientadas por eventos, as interações do utilizador são captadas e enviadas para as respetivas rotinas de tratamento. De que parte do sistema é a responsabilidade de execução da rotina principal de captação desses eventos?

- ☐ É do próprio Sistema Operativo (SO) do dispositivo móvel.
- ☐ É do processo da aplicação que contém as rotinas de tratamento do evento.
- ☐ É da própria rotina de tratamento do evento.

Q3.: As aplicações para SOs de dispositivos modernos são normalmente construídas usando a arquitetura MVC. Dentro desta arquitetura, em que componente é que se inserem as tais rotinas de processamento de um evento?

- ☐ No Modelo.
- ☐ No Continente.
- ☐ Na Visão.
- ☐ No Virtual.
- ☒ No Controlador.
- ☐ No Manipulador.

Q4.: Para aplicações Android™, a interface de utilizador é definida recorrendo a dois tipos de elementos gráficos configurados em ficheiros XML, como o que se apresenta a seguir:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    >
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:text="Este texto esta centrado?"
    />
</LinearLayout>
```

Qual o tipo de elemento gráfico `LinearLayout`?

- ☐ O `LinearLayout` é um objeto interativo (*widget*).
- ☒ O `LinearLayout` é um contentor (*container*).
- ☐ O `LinearLayout` é um consumidor (*Listener*).

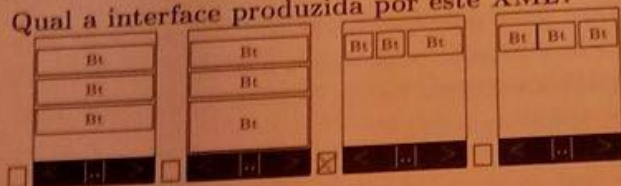
Q14.: Atente novamente no código XML incluído em cima. Considere que estava a utilizar a aplicação Android™ com a interface de utilizador definida por aquele ficheiro, num dispositivo com ecrã de 10". O que pode dizer acerca da disposição da frase Este texto está centrado? no ecrã?

- ☐ O texto está alinhado à esquerda.
☒ O texto está centrado.
☐ O texto está alinhado à direita.
☐ O texto nem aparece.

Q15.: Considere agora o excerto XML seguinte:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="horizontal"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    >
    <Button
        android:layout_width="0dp"
        android:layout_height="fill_parent"
        android:text="Bt"
        android:layout_weight="1" />
    <Button
        android:layout_width="0dp"
        android:layout_height="fill_parent"
        android:text="Bt"
        android:layout_weight="1" />
    <Button
        android:layout_width="0dp"
        android:layout_height="fill_parent"
        android:text="Bt"
        android:layout_weight="2" />
</LinearLayout>
```

Qual a interface produzida por este XML?



Plataforma e Aplicações Android™

Q5.: Como se chama a máquina virtual Java onde correm as aplicações Android™?

Dalvik

Q6.: Qual das seguintes está correta?

- ☐ É possível correr uma aplicação Android™ numa máquina virtual Java comum, mas não é possível correr uma aplicação Java comum na máquina virtual do Android™.
☐ Não é possível correr uma aplicação Android™ numa máquina virtual Java comum, tal como não é possível correr uma aplicação Java comum na máquina virtual do Android™.

- ☐ É possível correr uma aplicação Android™ numa máquina virtual Java comum e é possível correr uma aplicação Java comum na máquina virtual do Android™.
☒ Não é possível correr uma aplicação Android™ numa máquina virtual Java comum, mas é possível correr uma aplicação Java comum na máquina virtual do Android™.

Q7.: A pilha de software da plataforma Android™ está estruturada em 4 camadas. Das opções seguintes, quais não correspondem a designações para essas camadas?

- ☐ Camada das bibliotecas nativas.
☐ Camada do núcleo do sistema.
☒ Camada do ozono.
☐ Camada de *framework* de aplicações.
☐ Camada aplicacional.
☒ Camada de interface de utilizador.

Q8.: Em que camada da pilha de software é que se situam os gestores de recursos (e.g., de sensores, de atividades ou notificações) de um dispositivo móvel com Android™?

- ☐ Camada das bibliotecas nativas.
☐ Camada do núcleo do sistema.
☐ Camada do ozono.
☒ Camada de *framework* de aplicações.
☐ Camada aplicacional.
☐ Camada de interface de utilizador.

Q9.: É verdade que a plataforma Android™ inclui uma biblioteca chamada *bionic libc*?

- ☐ Não, não é verdade.
☒ Sim, é verdade.

Q10.: Quando se compila e empacota um projeto Android™, é automaticamente gerado um ficheiro chamado *R.java*, que contém apontadores (referências) que simplificam a implementação de uma aplicação. Em que diretoria é que este ficheiro é criado?

- ☐ *project_root/src* ☐ *project_root/res*
☒ *project_root/gen* ☐ *project_root/bin*

Q11.: Um dos recursos mais úteis aquando da depuração de aplicações Android™ é encabeçado pela classe *Log*. Qual o número (mínimo) de argumentos que os métodos *v()*, *i()*, *d()*, *w()* e *e()* aceitam?

- ☐ 0 ☐ 1 ☒ 2 ☐ 3 ☒ $\sqrt{1}$

Q12.: Qual dos métodos enunciados antes é o menos verboso (i.e., é normalmente usado para escrever menos informação no *log*)?

- ☒ *Log.v()* ☐ *Log.w()* ☐ *Log.i()* ☐ *Log.e()*

☐ Log.d()

Q13.: Qual o `import` que precisa colocar na implementação da aplicação para fazer uso das funcionalidades do `logcat`?

- ☒ `android.util.Log;` ☐ `feline.cat.Log;`
☐ `android.os.Log;` ☐ `java.lang.Log;`

Q16.: No contexto da plataforma Android™, o que significa o acrónimo ADB?

A Android D debug B ridge

Q17.: Qual das seguintes afirmações é verdadeira?

- ☐ Para correr num sistema Android™, todas as aplicações têm de ser obrigatoriamente assinadas digitalmente.
☐ Só as aplicações Android™ em fase de depuração é que não precisam ser assinadas digitalmente para correr num sistema Android™.
☐ As versões finais das aplicações Android™ não precisam ser assinadas digitalmente (só as que são compiladas com `> ant debug`).
☐ As aplicações Android™ não são assinadas digitalmente.

Q18.: Um programador sentiu a necessidade de implementar uma série de métodos, que usa em diversas componentes da sua aplicação, numa classe a que chamou `Utility`, tendo colocado o ficheiro `Utility.java` junto dos ficheiros que codificam a aplicação. Ao compilar, dá erro. Qual, ou quais, podem ser os motivos?

- ☐ O nome do `package` estará provavelmente mal no ficheiro `Utility.java`.
☐ O programador esqueceu-se de declarar a classe `Utility` no `AndroidManifest.xml`.
☐ O nome do ficheiro está mal.
☒ A diretoria onde o ficheiro foi colocado está errada.

A Componente Atividade

Q19.: A inclusão de uma Atividade numa aplicação Android™ é feita através da reescrita de um ou mais métodos da classe `Atividade`. Em qual ou quais dos métodos seguintes é obrigatória a invocação do método com o mesmo nome na superclasse aquando da sua reescrita?

- ☐ `onResume()` ☐ `onStart()` ☐ `onStop()`
☐ `onPause()` ☐ `onDestroy()` `onCreate()`
☐ Em nenhum dos anteriores.

Q20.: Recorde o ciclo de vida das Atividades numa aplicação Android™. Considere que um

utilizador já tinha uma aplicação em execução e uma Atividade em foco, e que efetuou os seguintes passos:

1. Carregou no botão `Home`;
2. Abriu outra aplicação e fartou-se;
3. Carregou no botão `Aplicações Recentes`, e voltou à aplicação/Atividade que estava a usar inicialmente.

Qual a sequência de métodos percorrida pela Atividade referida acima?

- ☐ `onCreate()` → `onStart()` → `onResume()` → `onPause()` → `onStop()` → `onDestroy()`
☐ `onPause()` → `onStop()` → `onDestroy()` → `onCreate()` → `onStart()` → `onResume()`
☐ `onPause()` → `onResume()`
☒ `onPause()` → `onStop()` → `onRestart()` → `onStart()` → `onResume()`

Q21.: Ainda sobre o ciclo de vida de uma Atividade, é possível que uma Atividade seja destruída pelo sistema sem passar pelo método `onDestroy()`?

- ☐ Não, nem pensar.
☒ Sim, é perfeitamente possível.

Intentos e Permissões

Q22.: Considere analisar o seguinte excerto de código Java e assuma que o Serviço `ServiceAlarms` estava, de facto, bem implementado e disponível:

```
public class FloatingAlarms extends Activity{
    ...
    public void onClick(View v){
        Intent oIntent =
            new Intent(this, ServiceAlarms.class);
        startService(oIntent);
    }
}
```

Qual o tipo de intento definido no excerto de código?

- ☐ Intento definido. ☐ Intento indefinido.
☐ Intento implícito. ☒ Intento explícito.

Q23.: Assuma que tudo o que estava para além do código que é mostrado no excerto anterior estava bem implementado. A compilação deste projeto será bem ou mal sucedida?

- ☐ Mal sucedida. ☐ Bem sucedida.

Q24.: Escreva o nome do método (e indique os argumentos) que lhe permite despoletar uma Atividade da qual aguarda um retorno:

`startActivityForResult()`
(valor extra)
↳ para a mão ser usada

Q25.: Quando um intento regressa com uma resposta, o método cujo protótipo se inclui a seguir é despoletado automaticamente na Atividade que o vai receber:

```
void onActivityResult(int requestCode, int resultCode, Intent data);
```

Qual o significado do requestCode?

- ☒ O requestCode é o parâmetro que contém o valor que é devolvido pela segunda Atividade.
- ☐ O requestCode é um inteiro que informa se a segunda Atividade terminou a tarefa para a qual foi despoletada com sucesso ou não.
- ☐ O requestCode é um inteiro que identifica a que intento corresponde este retorno específico, caso tenham sido começadas várias atividades com retorno.

Q26.: Quando uma Atividade devolve o retorno à Atividade que a invocou, esse retorno é devolvido no mesmo Intento que a despoletou?

- ☒ Sim.
- ☐ Não, é enviado num Intento diferente.
- ☐ Não, o retorno é transmitido via sistema. Não há mais intentos envolvidos.

Q27.: Considere que o Serviço ServiceAlarms está definido no manifesto da aplicação conforme se mostra a seguir:

```
<service android:name="ServiceAlarms">
  <intent-filter>
    <action android:name="pt.ubi.di.pmd.
      ServiceAlarms.SERVICE" />
  </intent-filter>
</service>
```

Já se sabe que, quando um intento é enviado, é o SO que resolve esse intento e o entrega à componente destino. A questão é: Para o intento definido no excerto de código anterior, o SO vai fazer uso do filtro de intents definido no manifesto?

- ☐ Os filtros de intents são sempre consultados.
- ☐ Não, neste caso não vai usar este filtro de intents.

Q28.: O que é que, de modo mais vincado, impede uma aplicação Android™ de aceder aos recursos de outra aplicação no sistema?

- ☐ O facto de cada aplicação correr numa máquina virtual diferente.
- ☐ O facto de a cada aplicação ser atribuído um identificador de utilizador (UID) diferente aquando da instalação, e de cada aplicação correr, portanto, num processo com um UID diferente de todas as outras aplicações.

☒ O facto de cada aplicação ter uma diretoria só para si, cifrada com uma chave diferente das diretorias das outras aplicações.

☐ O facto de cada aplicação fechar a porta antes de começar a executar, impedindo que as outras entrem.

Q29.: Quando se quer utilizar um recurso protegido pelo sistema Android™ é necessário pedir autorização explícita no manifesto da aplicação, incluindo um elemento semelhante ao seguinte:

```
<uses-permission android:name="perm_name"/>
```

Onde é que este elemento deve ser colocado?

- ☒ Dentro do elemento manifest, mas fora do elemento application.
- ☐ Dentro do elemento application, mas fora do elemento manifest.
- ☐ Dentro do elemento application, mas fora de qualquer elementos que defina uma componente.
- ☐ Dentro do elemento que define a componente que vai usar a permissão.

Gestão e Armazenamento de Dados

Q30.: Considere analisar o seguinte excerto de código, incluído na implementação de uma Atividade:

```
SharedPreferences oSP = getPreferences();
boolean bR = oSP.getBoolean("rec", false);
if ( bR )
  Toast.makeText(this, "Hello", Toast.
    LENGTH_SHORT).show();
else {
  SharedPreferences.Editor oEd = oSP.edit();
  oEd.putBoolean("rec", true);
}
```

Considere que nunca correu a aplicação depois da instalar. Quando é que a mensagem Hello vai aparecer no ecrã?

- ☒ Logo na primeira execução.
- ☐ Só na segunda execução.
- ☐ Só nas execuções número par.
- ☐ Só nas execuções número ímpar.
- ☐ Nunca.

Q31.: O que entende por *armazenamento externo*?

- ☒ É o nome dado a todo o armazenamento que é concretizado por cartões de memória que podem ser removidos do dispositivo móvel com sistema operativo Android™ e que, portanto, pode não existir em determinados dispositivos.
- ☐ É a parte do armazenamento do dispositivo móvel com sistema operativo Android™ que fica visível para outros dispositivos (móveis ou não) aquando da sua ligação via, e.g., cabo USB.

- ☐ É o nome dado a todos os recursos remotos de armazenamento, nomeadamente acessíveis via rede local ou Internet.
- ☐ É o nome dado à parte do armazenamento que contém os ficheiros de determinada aplicação, e que só pode, portanto, ser usado por essa aplicação.

Q32.: Qual o nome da classe que pode ser estendida para o ajudar na tarefa de criação e atualização consistente de uma base de dados numa aplicação Android™?

☐ SQLiteOpenHelper ☐ SQLiteHelper

☐ SQLiteDatabase ☐ SantosLittleHelper

Q33.: A classe mencionada na pergunta anterior disponibiliza 2 métodos (`onCreate()` e `onUpgrade()`). Assinale a, ou as, opções que correspondem a situações em que o método `onCreate()` é invocado automaticamente pelo Android™?

☐ Na primeira vez que a base de dados é criada.

☐ Sempre que a base de dados é aberta.

☐ Sempre que a base de dados é atualizada.

Q34.: No âmbito da depuração de aplicações com bases de dados SQLite3, foi-lhe sugerido o uso da ferramenta/shell `sqlite3`. Dentro da *shell*, qual era a instrução que lhe permitia ver a definição das tabelas da base de dados?

☐ `.show` ☐ `.make-breakfast` ☐ `.schema`

☐ `.tables` ☐ `.descriptions` ☐ `.databases`

Q35.: A manipulação de uma base de dados numa aplicação é normalmente conseguida recorrendo a instruções SQL, definidas como *Strings* dentro do código Java. Que nome se dá ao SQL, quando usado desta forma?

☐ Host Language ☐ Included SQL ☐ NoSQL

☐ Embedded SQL ☐ Guest SQL ☐ SSQ

Provedores de Conteúdos

Q36.: Se quiser implementar um Provedor de Conteúdos para uma base de dados SQLite3, um dos métodos a reescrever é o método `query()`. O que é que este método deve devolver?

☐ Deve devolver `null`.

☐ Deve devolver um objeto da classe _____

Q37.: O acesso a Provedores de Conteúdos em Android™ é normalmente feito através da especificação de um URI. Para o caso em análise, este URI pode ser tipicamente decomposto em 4 partes, esquematizadas da seguinte forma:

parte-1://parte-2/parte-3/parte-4

Selecione uma opção em cada uma das colunas da tabela seguinte de forma a indicar o que é que deve estar na parte-1 e na parte-2 do URI para um Provedor de Conteúdos:

parte-1	parte-2
<input type="checkbox"/> http	<input type="checkbox"/> nome de uma tabela
<input type="checkbox"/> content	<input type="checkbox"/> nome da autoridade
<input type="checkbox"/> provider	<input type="checkbox"/> id da linha a devolver
<input type="checkbox"/> uri	<input type="checkbox"/> android

A Componente Serviço e Threads

Análise o excerto de código Java seguinte e responda às questões que se lhe seguem:

```
public class ServiceAlarms extends Service {
    private final IBinder oBinder = new
        BinderLocal();

    public class BinderLocal extends Binder {
        ServiceAlarms getService() {
            return ServiceAlarms.this;
        }
    }

    @Override
    public int onStartCommand(
        Intent intent, int flags, int startId) {
        Toast.makeText(this, "Service on!", Toast.
            LENGTH_SHORT).show();
        new Thread(){
            public void run(){
                try{
                    Thread.sleep(5000);
                }catch( InterruptedException oIE ){
                    oIE.printStackTrace();
                }
            }
        }.start();
        return START_STICKY;
    }

    @Override
    public IBinder onBind(Intent intent) {
        return oBinder;
    }
}
```

Q38.: Dada a sua implementação, o `ServiceAlarms` é um Serviço *started* (sem vínculo) ou *bound* (com vínculo)?

- ☐ Pode ser invocado de ambas as formas. ☐ *Started*
- ☐ Nem um nem outro. ☐ *Bound*

Q39.: Quando é que a mensagem `Service on!` é exibida no ecrã do dispositivo móvel?

- ☐ Quando uma aplicação chama o Serviço com o método `startService()`.
- ☐ Quando uma aplicação chama o Serviço com o método `bindService()`.

- ☐ Nunca.
- ☐ Logo que a primeira toast mista estiver pronta.

Q40.: Qual a duração de uma Toast definida com `Toast.LENGTH_SHORT`?

- ☐ 2 segundos
- ☐ 1,5 segundos
- ☒ 2 segundos
- ☐ 2,5 segundos
- ☐ 3 segundos
- ☐ 3,5 segundos

Q41.: Como classifica a afirmação seguinte?

Por defeito, um Serviço corre no mesmo processo e thread que trata da interface de utilizador de uma aplicação Android™.

- ☐ A afirmação é verdadeira.
- ☐ A afirmação é falsa.

Q42.: O que significa o valor `START_STICKY`?

- ☒ O valor devolvido define que, caso o Serviço seja morto pelo sistema, este o deve tentar recomençar quando tiver recursos.
- ☐ O valor devolvido define que, caso o Serviço seja morto pelo sistema, este não o deve tentar recomençar quando tiver recursos.
- ☐ O valor devolvido define que o sistema deve evitar terminar este Serviço.
- ☐ Significa que este Serviço é peganhoso. Yeach!

Q43.: Qual das seguintes afirmações é verdadeira?

- ☐ Um Serviço sem vínculo termina logo que todo o trabalho definido no método `onStartCommand()` termine.
- ☒ Um Serviço com vínculo termina logo que todas as componentes que a si estavam ligadas terminem o vínculo.
- ☐ Um Serviço com vínculo tem de ser terminado com o método `stopSelf()`.

Q44.: No jargão associado ao Java, que nome se dá à classe criada com `new Thread(){...}`?

- ☐ Interface.
- ☐ Classe abstrata.
- ☐ Pseudo classe.
- ☐ Classe anónima.

Recetores de Difusão

Q45.: É verdade que, para alguns eventos, só se podem registar os respetivos Recetores de Difusão dinamicamente (i.e., durante a execução da aplicação)?

- ☐ Sim, é verdade!
- ☐ Não, qualquer Recetor de Difusão pode sempre ser registado através do `AndroidManifest.xml`.

Q46.: Para se criar um Recetor de Difusão é preciso estender a classe `BroadcastReceiver`. Selecciona, de seguida, o método que é necessário

implementar para ter uma concretização deste componente?

- ☐ `onStart()`
- ☐ `onIntent()`
- ☒ `onReceive()`
- ☐ `onBroadcast()`

Q47.: Os intents em difusão podem ser enviados através de dois métodos. Qual deles garante que todos os recetores registados para esses intents o vão receber?

- ☐ O método `sendBroadcast()`, já que os que são enviados através de `sendOrderedBroadcast()` podem ser cancelados por algum dos recetores.
- ☐ O método `sendOrderedBroadcast()`, já que os que são enviados através de `sendBroadcast()` podem ser cancelados por algum dos recetores.
- ☐ Ambos os métodos dão essa garantia.

Sensores

Q48.: A *Framework* de Sensores do Android™ é constituída por X classes/interfaces. Qual é a interface que deve ser implementada para obter valores de um sensor?

- ☐ `SensorManager`
- ☐ `Sensor`
- ☐ `SensorReceiver`
- ☐ `SensorIntent`
- ☐ `SensorEvent`
- ☐ `SensorEventListener`

Q49.: Na pergunta anterior falta um número. Que número deve X tomar?

- ☐ X = 1
- ☐ X = 2
- ☐ X = 3
- ☐ X = 4
- ☐ X = 5
- ☐ X = y/2 - 3, em que y = 18

Q50.: A interface referida em cima obriga à implementação de dois métodos para a obtenção de valores de um sensor. Que métodos são esses?

- ☐ `onAccuracyChanged()`
- ☐ `onNewValue()`
- ☐ `onMovement()`
- ☐ `onSensorValue()`
- ☐ `onSensorChanged()`
- ☐ `getValue()`