



Programação de Dispositivos Móveis
Exame de Época Normal – Tipo 01
Licenciatura em Engenharia Informática
Departamento de Informática
Universidade da Beira Interior

Duração da Prova: 1h40

22 de Janeiro de 2015

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.: O que tem a dizer acerca da seguinte afirmação?

A definição de dispositivos móveis dita que estes têm de obrigatoriamente possuir capacidade de processamento, tal como integrar dispositivos de entrada e saída.

- ☐ Esta afirmação está totalmente correta.
- ☐ A primeira metade da afirmação está correta. A segunda metade é falsa.
- ☐ A primeira metade da afirmação está errada. A segunda metade é verdadeira.
- ☐ Esta afirmação está totalmente errada.

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 situa a implementação dos objetos e dos métodos que capturam o comportamento base de

uma aplicação?

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

Q4.: Para aplicações Android™, a interface de utilizador é definida recorrendo a dois tipos de elementos gráficos, tipicamente configurados em ficheiros XML. Nas opções seguintes, selecione todas as que se referem a elementos do tipo contentor (container):

- ☐ EditText
- ☐ LinearLayout
- ☐ ScrollView
- ☐ TextView
- ☐ Button
- ☐ RelativeLayout

Q5.: Observe atentamente o excerto XML seguinte:

```
<?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:layout_gravity="center"
        android:text="Este texto esta centrado?"
    />
</LinearLayout>
```

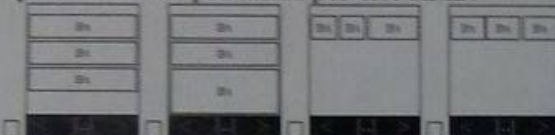
Considere que estava a utilizar a aplicação Android™ com a interface de utilizador definida por aquele ficheiro, num dispositivo com ecrã de 10". O que pode dizer acerca da disposição da frase Este texto esta centrado? no ecrã?

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

Q6.: 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="vertical"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    >
    <Button
        android:layout_width="fill_parent"
        android:layout_height="0dp"
        android:text="B1"
        android:layout_weight="1" />
    <Button
        android:layout_width="fill_parent"
        android:layout_height="0dp"
        android:text="B2"
        android:layout_weight="1" />
    <Button
        android:layout_width="fill_parent"
        android:layout_height="0dp"
        android:text="B3"
        android:layout_weight="2" />
</LinearLayout>
```

Qual a interface produzida por este XML?



Plataforma e Aplicações Android™

Q7.: Qual das seguintes afirmações está correta?

- ☐ É possível definir as coordenadas geográficas do GPS dos dispositivos virtuais Android™ disponibilizados com o SDK Android™.
- ☐ Não é possível definir as coordenadas geográficas do GPS dos dispositivos virtuais Android™ disponibilizados com o SDK Android™.

Q8.: Por curiosidade, como se expande o acrónimo GPS? Brincadeira! A verdadeira pergunta é: Como se expande o acrónimo SDK?

S _____

D _____

K _____

Q9.: 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™.

Q10.: Em que camada da pilha de software da plataforma Android™ é que se situa a implementação do SQLite3?

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

Q11.: Quando se compila e empacota um projeto Android™, são automaticamente gerados arquivos com extensão .apk. Em que diretoria é que esses arquivos ficam guardados?

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

Q12.: Um dos recursos mais úteis aquando da depuração de aplicações Android™ é encabeçado pela classe Log. Cada um dos métodos v(), i(), d(), w() e e() que disponibiliza aceita dois parâmetros. Quais os tipos/classes desses parâmetros?

Parâmetro 1	Parâmetro 2
<input type="checkbox"/> int	<input type="checkbox"/> int
<input type="checkbox"/> long	<input type="checkbox"/> long
<input type="checkbox"/> String	<input type="checkbox"/> String
<input type="checkbox"/> double	<input type="checkbox"/> double
<input type="checkbox"/> Intent	<input type="checkbox"/> Intent

Q13.: 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()

Q14.: 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;

Q15.: A adb é uma das ferramentas mais úteis aquando da implementação e depuração de aplicações Android™. Em que diretoria do SDK é que a ferramenta pode ser encontrada?

- ☐ build-tools ☐ system-images ☐ temp
- ☐ platforms ☐ android ☐ etc
- ☐ platform-tools

Q16.: Outra ferramenta bastante interessante fornecida com o SDK é a que integra, e.g., o visualizador de *logcat*, gestor de ficheiros dos dispositivos móveis, visualizador de hierarquia, etc. Como se chama essa ferramenta?

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 `-jarsigner`).
- ☐ 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.: Recorde o ciclo de vida de uma Atividade. Em qual das seguintes situações é que o ciclo de vida de uma Atividade pode passar pelo método `onPause()` sem passar pelo método `onStop()`?

- ☐ Em qualquer situação. Os dois métodos concretizam funcionalidades disjuntas pelo que, se passar num deles, não pode passar no outro.
- ☐ Quando uma caixa de diálogo se coloca em primeiro plano, deixando a atividade principal desfocada, mas visível.
- ☐ Em nenhuma situação. O ciclo de vida de uma atividade determina que se uma Atividade passar pelo método `onPause()`, também terá de passar pelo método `onStop()`.

Q20.: 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()`
- ☐ Em nenhum dos anteriores.

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

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

Intentos e Permissões

Q22.: Qual dos seguintes excertos de código Java define um intento implícito?

- ☐

```
Intent i1 = new Intent(this, nextAct.class);
```
- ☐

```
Intent i1 = new Intent(Intent.ACTION_SEND);
```
- ☐

```
Intent i1 = new Intent();  
i1.setComponent(new ComponentName("com.  
android.calculator2", "com.android.  
calculator2.Calculator"));
```

Q23.: Qual dos seguintes métodos lhe permite despoletar uma Atividade da qual aguarda um retorno?

- ☐ `startActivityForResult(Intent, int)`
- ☐ `startActivityWithResult(Intent)`
- ☐ `startActivityForResult(Intent, int)`

Q24.: 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.

Q25.: Quando uma Atividade devolve o retorno à Atividade que a invocou, esse retorno é devolvido no mesmo `Intent` que a despoletou?

- ☐ Sim. ☐ Não, é enviado num `Intent` diferente.
- ☐ Não, o retorno é transmitido via sistema. Não há mais intents envolvidos.

Q26.: Considere que um utilizador instala duas aplicações de um mesmo programador, executando ambas no seu dispositivo móvel com Android™. Nesta situação, qual das seguintes frases está correta?

- ☐ As duas aplicações correm na mesma máquina virtual Java, com o mesmo `User ID` (UID).
- ☐ As duas aplicações correm em máquinas virtuais Java diferentes, mas com o mesmo UID.
- ☐ As duas aplicações correm com UIDs diferentes em duas máquinas virtuais distintas.
- ☐ As duas aplicações correm com UIDs diferentes na mesma máquina virtual Java.

Q27.: Quando se quer utilizar um recurso protegido pelo sistema Android™ é necessário pedir autorização explícita no manifesto da aplicação, incluindo um elemento seme-

Ibante 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 elemento que defina uma componente.
- ☐ Dentro do elemento que define a componente que vai usar a permissão.

Gestão e Armazenamento de Dados

Q28.: Considere analisar atentamente o seguinte excerto de código, incluído na implementação de uma Atividade:

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

Considere que nunca correu a aplicação depois da instalar mas que agora a executava 3 vezes seguidas. Qual a sequência de mensagens que iria ver no ecrã?

- ☐ Bonjour → Bonjour → Bonjour
- ☐ Au Revoir → Hasta la Vista → Baby
- ☐ Hello → Bonjour → Hello
- ☐ Bonjour → Hello → Bonjour
- ☐ Hello → Hello → Hello

Q29.: Um programador pedia-lhe a sua opinião de especialista relativamente ao armazenamento em Android™. Basicamente, queria guardar um objeto da classe Pessoa que ele próprio tinha implementado. Sugeria-lhe a utilização das SharedPreferences para este efeito?

- ☐ Sim, sugeria.
- ☐ Não, não sugeria.

Q30.: 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.

Q31.: 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

Q32.: 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.

Q33.: 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

Q34.: 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
- ☐ SSQL

Provedores de Conteúdos

Q35.: 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 _____

Q36.: 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

Q37.: Qual é a *tag* XML usada para definir um Provedor de Conteúdos no AndroidManifest.xml?

- ☐ <content>
- ☐ <content-provider>
- ☐ <muito-content>
- ☐ <trist>
- ☐ <provider>

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_LONG).show();
        Thread.sleep(5000);
        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_LONG*?

- ☐ 1 segundo ☐ 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.: Na aula que aborda a componente Serviço é mencionado o facto de um Serviço ser normalmente um bom candidato à destruição quando há falta de recursos. Colocar os Serviços em primeiro plano com o método *startForeground(...)* pode diminuir a probabilidade de serem destruídos. O método aceita 2 parâmetros. Qual das seguintes classes define um desses parâmetros?

- ☐ *Intent* ☐ *Service*
☐ *Notification* ☐ *Handler*

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*. Selecione, de seguida, o método que é necessário implementar para ter uma concretização deste componente?

- ☐ *onStart()* ☐ *onIntent()*
☐ *onReceive()* ☐ *onBroadcast()*

Q47.: Os intentos em difusão podem ser enviados através de dois métodos. Qual deles garante que todos os recetores registados para esses intentos 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.: Por quantas classes/interfaces é constituída a *Framework* de Sensores da plataforma Android™?

- ☐ Por um número inusitado de classes/interfaces.
☐ 0 ☐ 1 ☐ 2 ☐ 4 ☐ 8

Q49.: A *Framework* de Sensores do Android™ contém uma classe chamada *SensorEvent*. É verdade que esta classe não disponibiliza quaisquer métodos de im-

plementação própria?

☐ Sim, é verdade.

☐ Não, é mentira.

Q50.: A classe referida na questão anterior contém um atributo chamado `values`, que pode ser usado para obter os valores devolvidos por determinado sensor de um dispositivo móvel. Qual é o tipo deste atributo?

☐ `long` ☐ `long[]` ☐ `double`

☐ `float[]` ☐ `int` ☐ `int[]`