

修平技術學院

資訊網路技術系專題報告

SCJP6.0 證照

組長：BN95021 許起源

組員：BN95022 蔡昀達

BN95029 張晏誠

BF95050 鄭旭淳

指導老師：_____ 老師

口試老師：_____ 老師

_____ 老師

_____ 老師

99年1月

摘要

為了能畢業後順利進入資訊科技工作領域，當社會新鮮人進入職場，能拿什麼跟別人競爭？相信一件事：如果有一張(最好數張以上)證照，除了對自己能力可以提出有力「證明」，也會從眾多求職者脫穎而出，成為面試主管審慎考慮的對象。

Java 以其平台無關性、面向對象、支持多線程等優點成為越來越多的程序開發人員的新寵，Java 技術以其獨特的優勢在越來越多的領域得以使用和發展。而各大企業競相爭取懂 Java 技術的人才，也使 Java 專業認證成為 IT 人員最想取得的一張證書。

目錄

摘要	2
目錄	3
第一章 簡介.....	4
第二章 參考資料.....	5
第三章 報考程序及注意事項.....	7
第四章 讀書計畫.....	9
第五章 考題詳解.....	11
第六章 心得感想.....	55
第七章 附錄	58

簡介

自從 1995 年 Sun 提出 Java 技術與解決方案開始，至今 Java 已是企業在開發重要專案與系統時所選擇的技術。因此不管是未來從事任何相關的 IT 職務，如程式設計師、網頁開發師、系統分析師或是 J2EE 系統架構師等，都必須充分了解並熟悉運用以 Java 程式語言為基礎的系統開發技術。而 SCJP 證照所認證的技術，正是這項技術。在 Java 認證範疇中，若想再考取其它進階認證（例如 SCWCD、SCJD、SWBCD 等）之前，都必須先取得 SCJP，因此，SCJP 認證等同於 Java 領域的「基本學歷」。

參考資料

Java 認證 SCJP 6.0/5.0--猛虎出關

本書支援 310-065/066 Java SCJP 6.0 認證考試，內容涵蓋 Java 基礎知識與認證考試範圍，為您建立學習的正確觀念，輕易掌握關鍵考題。SCJP 6.0 猛虎出關?讓您成為 Java 界的老虎伍茲!經過了 13 年發展，1995 才問世的 Java 從一個默默無名的技術，不斷地拓展並滲透到每一個人的生活中，小到生活中的晶片卡、大哥大、機上盒 (set-top box)、個人電腦，大到企業組織系統乃至美國太空總署火星計畫，都有它的蹤跡。根據專家預測，Java 技術今後將會更廣泛地被使用在企業應用、桌面端應用和行動應用這三個層面上。非官方調查也顯示，目前全球已有超過 61% 的軟體發展人員使用 Java 語言，2008 年這個數據更將上升至 70%。當今 Java 開發程式師在全球已超過 300 萬人，有 7.67 億支以上的電話支援 Java 技術，更有超過 10 億張 Java 卡在世界各地被配置。您還在等甚麼呢?歡迎一同進入 Java 的行列，為世界探索更多新的可能!

SCJP 6.0 認證教戰手冊

1. SCJP 認證考試介紹：詳細介紹 SCJP 6.0 認證考試的範圍與考試題型，並一步一步地帶領讀者在 Prometric 官方網站上註冊個人資料與訂定考場。列出 Prometric 在台灣所有合法考場的名稱與地址。

2. Java 基本語法介紹：以日常生活所熟悉的事物當作範例來說說 Java 基本語法，完全不需要強記。

3. Java 物件導向觀念說明：作者以業界專案設計的經驗來說明物件導向的觀念。例如抽象類別、介面由系統分析師來撰寫的原因。

4. Java 重要類別庫說明與應用：將 SCJP 認證考試所涵蓋的類別庫及其重要方法都列表加以說明，並將應用方式以範例完整呈現。

5. 244 題完全擬真試題詳解：保證符合 SCJP 6.0 的考試範圍與命題方向，並將考題分析與解題技巧錄製成『影音光碟』，讓讀者即使在家，也能夠隨時聽講，以熟練考試題型而順利取得 SCJP 國際證照。

報考程序

由於系上主任和老師們積極地為同學們籌劃一系列國際專業證照課程,我們藉著這個機會,參加了 SCJP 認證,報名時間是 2009 年 10 月 29 號,報名方式採網路報名,網頁全部都是英文介面,系上網站都有報名教學,報名費是 2950 元整,因為是團體報名所以有優惠,填寫完報名表連同報名費一起繳交給系助理,再影印身分證正反副本,即完成報名手續。

注意事項

1. 考前給自己足夠的信心。通過做練習和模擬軟件測驗
2. 按預約的時間提前到達考試中心，熟悉考試環境，穩定情緒
3. 請記得攜帶身分證明文件，以方便證明身分
4. 正式考試前的問卷調查，給定時間為 15 分，有 10 來個問題，給出四個選項供你選擇，主要調查你對 SCJP 要考查的知識點的掌握情況
5. 正式考試，共 120 分鐘，選擇題在題面中註明了正確答案的數目（如：choose two）看清楚再作答，有很相似的題型

讀書計畫

(BN95021)許起源

在學期間因為學過 JAVA 課程,所以我們首先拿出以前的課本出來複習,暑假也有安排 SCJP 證照班,加上指導老師有安排 1 個月的課程指導與交流時間,最後的一個月我們用來做最後衝刺,一起合購了 SCJP 題庫研讀,總共 358 題,利用課餘時間熟記題庫,並且利用軟體測驗自己是否完全記熟。

(BN95022)蔡昀達

知道要考試之後,每天除了把在學校學過的東西再復習外,也配合著學校暑期進度一起練習,在自習與上課之間能雙管齊下;也多利用在每次的討論時間,把一禮拜所遇到的問題盡量提出,使成長的不是只有自己,而是整個團隊一起進步,在往後的考試上也無往不利。

(BN95029)張晏誠

就在一個美麗的早晨，我深呼吸，在心理勵志自己，今年一定要考上一張證照，就決定是 scjp 了，於是我開始去學校的圖書館找資料，把歷屆的試題找了出來，看了幾題發現我的基本功還不夠扎實，於是我又從書櫃裡把塵封已久的 java 課本給拿了出來，每個晚上，java 就像我的新任女友，一直陪著我直到天明，然後喝過一杯咖啡後，漫長的一天就換歷屆試題陪著我，日復一日，年過一年。

(BN95050)鄭旭淳

新的一天，新的開始，每個人每件事都要有一定的規劃，就像吃飯也是有規劃的，

每個人都會早、中、晚，都吃一次飯一樣，都是要規劃的，讀書也是一樣，

今天可以加入考 scjp 這一組國證照組，如果沒有一定的讀書規劃，那麼多歷屆考試，那麼多基本題型，那麼多前人失敗的例子，在茫無頭緒日子你，就在等這你、我、他，要就一次考過，要不然就不要加入 scjp 以免浪費錢，也浪費每週老師帶我們復習的時間，所以我的讀書規劃跟計畫，一天 100 題歷屆考題~10 天就有 1000 題，等到考試那天我已經看完幾萬題了，已經夠我上考場了，這就是我成功的例子。

考題詳解

- 1.
12. void start() {
13. A a = new A();
14. B b = new B();
15. a.s(b);
16. b = null ;
17. a = null ;
18. System.out.println("start completed") ;
19. }

建立於第 14 行的 B 物件何時符合資源回收條件？

- A. 在第 17 行之後
- B. 在第 16 行之後
- C. 在第 18 行之後(方法結束後)
- D. 無法完全確定。
- E. 此物件不符合資源回收的條件。

解：15 行 a.s(b)把 b 的物件參考當成參數，我們不知道 a.s()這個方法做了什麼事，會不會再把物件參考 b 再給別人…，

Answer=D

2

1. class Test {
2. private Demo d ;
3. void start () {
4. d = new Demo();
5. this.takeDemo(d);
6. }
- 7.
8. void takeDemo(Demo demo) {
9. demo = null ;
10. demo = new Demo();
11. }
12. }

建立於第 4 行的 Demo 物件何時符合資源回收條件？

- A. 在第 5 行之後
- B. 在第 9 行之後
- C. 在 start()方法完成之後
- D. 在 takeDemo()方法完成之後

E. 在執行此程式的實體符合資源回收條件時

Answer=E

3

```
1. class Bar { }
1. class Test {
2. Bar doBar(){
3. Bar b = new Bar();
4. return b ;
5. }
6. public static void main (String args[]) {
7. Test t = new Test();
8. Bar newBar = t.doBar();
9. System.out.println( "newBar" );
10. newBar = new Bar();
11. System.out.println( "finishing" );
12. }
13. }
```

建立於第 3 行的 Bar 物件何時符合資源回收條件？

- A. 在第 8 行之後
- B. 在第 10 行之後
- C. 在第 4 行之後，即 doBar() 完成時
- D. 在第 11 行之後，即 main() 完成時

解：第 3 行建出的物件一直先由 doBar() 方法中的 b 存取，doBar() 方法結束後，b 也會消失，此時該物件的參考被傳出來給第 8 行中的 newBar 參考變數，由這個變數繼續參考著該物件，直到第 10 行，newBar 變數改指向新產生的物件，此時舊物件(第 3 行建的)就沒有參考指向它…，因此從這個時間點，這個物件就可

以被回收。

Answer=B

4

1. public class Foo {
2. public void main (String []args) {
3. System.out.println("Hello" +args[0]);
4. }
5. }

若此程式與以下命令列執行，結果為何？

Java Foo world

- A. Hello
- B. Hello Foo
- C. Hello world
- D. 編譯失敗
- E. 程式未執行

解：這個 main 不是標準的 main
public static void main(…

Answer=E

5

1. public interface Foo {
2. int k = 4 ;
3. }

下列哪三項等於第 2 行？(選擇三項)

- A. final int k = 4;
- B. public int k = 4;
- C. static int k = 4;
- D. abstract int k = 4;
- E. volatile int k = 4;
- F. protected int k = 4;

解：介面(本身是一個抽象的類別)中的變數隱含是 public、final、static(不必先有物件即可使用的變數)

Answer=ABC

6 下列哪兩項是介面定義內的有效宣告?(選擇兩項)

- A. void methoda();
- B. public double methoda();
- C. public final double methoda();
- D. static void methoda(double d1);
- E. protected void methoda(double d1);

解：介面中的方法是 public 的，非 static 的，非 final 的
_Answer=AB

7

- 1. // Point X
- 2. public class Foo {
- 3. public static void main (String [] args) throws Exception {
- 4. java.io.PrintWriter out = new java.io.PrintWriter (
- 5. new java.io.OutputStreamWriter(System.out), true);
- 6. out.println("Hello");
- 7. }
- 8. }

在第 1 行的 PointX 處，必須具有何項述句方能使此程式編譯?

- A. 無須任何述句。
- B. import java.io.*;
- C. include java.io.*;
- D. import java.io.PrintWriter;
- E. include java.io.PrintWriter;

解：不必 import 任何 package，題目中都是用全名的方式指定 java.io 包裝中的類別，include 是 c/c++的用法

Answer=A

8

```
1. // Point X
2. public class Foo {
3.     public static void main (String [] args) throws Exception {
4.         PrintWriter out = new PrintWriter (new
5.             java.io.OutputStreamWriter (System.out), true);
6.         out.println( "Hello" );
7.     }
8. }
```

在第 1 行的 PointX 處，必須具有何項述句方能使此程式編譯？

- A. 無須任何述句。
- B. `import java.io.PrintWriter;`
- C. `include java.io.PrintWriter;`
- D. `import java.io.OutputStreamWriter;`
- E. `include java.io.OutputStreamWriter;`

解：java.io.PrintWriter 必須被 import 進來，OutputStreamWriter 使用全名就不必 import

Answer=B

9

```
11. public static void main (String []args) {
12.     Object obj = new Object() {
13.         public int hashCode() {
14.             return 42 ;
15.         }
16.     }
17.     System.out.println(obj.hashCode());
18. }
```

結果為何？

- A. 42
- B. 在執行時期丟出一個例外。
- C. 由於第 12 行出現錯誤，編譯失敗。
- D. 由於第 16 行出現錯誤，編譯失敗。
- E. 由於第 17 行出現錯誤，編譯失敗。

Answer=A

10

```
1. public interface Test {  
2. int frood = 42 ;  
3. }  
4. class TestImpl implements Test{  
5. public static void main (String []args) {  
6. System.out.println(++frood);  
7. }  
8. }
```

結果為何？

- A. 0
- B. 1
- C. 42
- D. 43
- E. 編譯失敗。
- F. 在執行時期丟出一個例外。

解：frood 是 final static，也就是 frood 是一個常數的屬性，不可以在程式中改變其值(第 6 行)

Answer=E

11

```
1. public interface Test {  
2. int frood = 42 ;  
3. }  
4. class TestImpl implements Test{  
5. private static int frood;  
6. public static void main (String []args) {  
7. System.out.println(++frood);  
8. }  
9. }
```

結果為何？

- A. 0
- B. 1
- C. 42
- D. 43
- E. 編譯失敗。
- F. 在執行時期丟出一個例外。

解：第 7 行的 frood 是找 main 方法中所在類別中的 frood，若找不到才往父類

別找 frood，這題改變的

frood 不是 final...

Answer=B

12

```
1. public class Foo {
2. public static void main (String[] args) {
3. String s;
4. System.out.println( "s = " + s);
5. }
6. }
```

結果為何？

- A. s =
- B. s = null
- C. 編譯失敗。
- D. 在執行時期丟出一個例外。

解：在方法中的變數是不會“自動”給定預設初始值，第 3 行中定義了一個字串物件參考 s，第 4 行要

使用這個 s 變數必須要有值...(在使用任何變數之前，該變數必須先確認有值)

Answer=C

13

```
1. public class Foo {
2. static String s ;
3. public static void main (String []args) {
4. System.out.println( "s = " + s);
5. }
6. }
```

結果為何？

- A. s =
- B. s = null
- C. 在執行時期丟出一個例外。
- D. 由於第 2 行出現錯誤，編譯失敗。
- E. 由於第 4 行出現錯誤，編譯失敗。

解：類別的變數系統會給初值，物件的參考就是 null

Answer=B

14 下列何項方法必須由實作 java.lang.Runnable 介面的類別予以定義？

- A. void run()
- B. public void run ()
- C. public void start ()
- D. void run (int priority)
- E. public void run (int priority)
- F. public void start (int priority)

Answer=B

15 下列哪兩項有效的方法會在介面宣告中留有標記?(選擇兩項)

- A. void method1() ;
- B. public void method2() ;
- C. protected void method3() ;
- D. final public void method4() ;
- E. static public void method5() ;

解：介面中的方法是 public 的，非 static 的，非 final 的

. Answer=AB

16

- 1. interface I0{
- 2. public int method1(int x , int y);
- 3. public long method1(long x , long y);
- 4. }

下列何項可以編譯？

- A. interface I1 extends I0 { }
- B. interface I1 implements I0 { }
- C. abstract class C1 extends I0{
public int method1(int x , int y) { return 0 ; \ }
}
- D. abstract class C1 implements I0{
public int method1(int x , int y) { return 0 ; \ }
public short method1(long x , long y) { return 0 ; \ }
}
- E. class C1 implements I0{
public int method1(int x , int y) { return 0 ; \ }
public short method1(long x , long y) { return 0 ; \ }
}

解：

類別不可以繼承介面，只可以實作介面(可以同時好幾個)

介面不可以實作介面，只可以繼承介面(可以同時好幾個)

類別只能繼承單一類別

如果一個類別實作了一個介面，則類別必須提供介面中所有方法的實作

抽象類別亦可以實作一個介面，若不提供介面中所有方法的實作，則那些沒有被提供實作(透過 override)

的方法，需要再一次宣告該方法是抽象的…

Answer=A

17

```
1. public class Test {
2.     public static void main (String []args) {
3.         String foo = args [1];
4.         String bar = args [2];
5.         String baz = args [3];
6.         System.out.println( "baz = " + baz);
7.     }
8. }
```

從以下命令列啟動：

```
Java Test red green blue
```

結果為何？

- A. baz =
- B. baz = null
- C. baz =blue
- D. 編譯失敗。
- E. 在執行時期丟出一個例外。

解：陣列的索引從 0 開始，上面的命令列代入 3 個參數，索引為 0/1/2

Answer=E

18

```
1. public class Test {
2. public static void main (String []args) {
3. String foo = args [1];
4. String bar = args [2];
5. String baz = args [3];
6. System.out.println( "baz = " + baz);
7. }
8. }
```

輸出：

baz = 2

從下列命令列啟動會產生此輸出？

- A. java Test 2222
- B. java Test 1 2 3 4
- C. java Test 4 2 4 2
- D. java Test 4 3 2 1

解：baz 指向索引為 3 的元素

Answer=C

19

```
1. public class Test {
2. public static void main (String args []) {
3. for ( int i = 0 ; i <4 ; i ++ ) {
4. System.out.println(args[ i ]+ " " );
5. }
6. }
7. }
```

從以下命令列啟動：

Java Test w x y z

結果為何？

- A. w x y z
- B. Test w x y
- C. Test w x y z
- D. 編譯失敗。
- E. 在執行時期丟出一個例外。

解：索引沒有問題

Answer=A

20

```
1. public class Test {  
2. public static void main (String args ) {  
3. for ( int i = 0 ; i <3 ; i ++ ) {  
4. System.out.println(args + " " );  
5. }  
6. }  
7. }
```

從以下命令列啟動：

```
Java Test w x y z
```

結果為何？

- A. w x y z
- B. Test w x y
- C. Test w x y z
- D. 編譯失敗。
- E. 程式未執行。

解：第 2 行的 main 不是標準的 main，參數列必須是(String[] args) 沒有 main 就沒有程式的進入點

Answer=E

21 下列何項為有效的識別碼？

- A. break
- B. default
- C. _object
- D. a-classt

Answer= C

22 假定：

```
1. public class Test {  
2. public static void main (String [] args ) {  
3. unsigned byte b = 0 ;  
4. b--;  
5. System.out.println( "b= " +b );  
6. }  
7. }
```

結果為何？

- A. b=-1
- B. b=255
- C. b=127
- D. 編譯失敗。
- E. 在執行時期丟出一個例外。

解：第 3 行錯，unsigned 不是 java 的保留字

Answer=D

23 下列哪兩項是 Java 程式語言的保留字?(選擇兩項)

- A. goto
- B. strict
- C. include
- D. strictfp
- E. synchronize synchronized

Answer=AD

24

```
1. public class Test {  
2. public static void main (String [] args ) {  
3. String str = NULL;  
4. System.out.println(str );  
5. }  
6. }
```

結果為何？

- A. NULL
- B. 編譯失敗。
- C. 程式執行後未產生輸出。
- D. 在執行時期丟出一個例外。

解：第 3 行錯，NULL 應為 null...

Answer=B

25

```
11. public String switch (String str ) {  
12. String newStr = "" ;  
13. for ( int i = str.length() -1; i >=0; I --){  
14. newStr +=str.charAt( i );  
15. }  
16. return newStr;  
17. }  
18. public static void main (String [] args ) {  
19. System.out.println(switch( "ABCD" ) );  
20. }
```

結果為何？

- A. ABCD
- B. DCBA
- C. 編譯失敗。
- D. 在執行時期丟出一個 NullPointerException 例外。
- E. 在執行時期丟出一個 ArrayIndexOutOfBoundsException 例外。

解：switch 是保留字，不能用來做方法的命名

Answer=C

26 下列何項為關鍵字？

- A. go
- B. NULL
- C. Break
- D. Class
- E. transient

Answer=E

27 下列何項為關鍵字？

- A. TRUE
- B. array
- C. native
- D. reference
- E. byreference

Answer=C

Answer=B

28 假定：

- 11. `int index = 1;`
- 12. `String [] test = new String[3]`
- 13. `String foo = test[index] ;`
- 14. `System.out.println(" foo = " + foo) ;`

結果為何？

- A. `foo =`
- B. `foo = null`
- C. 編譯失敗。
- D. 在執行時期丟出一個例外。

解：第 12 行建出一個字串參考的物件，元素值被初始成 `null`

Answer=D

29

- 11. `int index = 1;`
- 12. `boolean [] test = new boolean [3]`
- 13. `boolean foo = test[index] ;`
- 14. `System.out.println(" foo = " + foo) ;`

結果為何？

- A. `foo = 0`
- B. `foo = null`
- C. `foo = true`
- D. `foo = false`
- E. 編譯失敗。
- F. 在執行時期丟出一個例外。

解：boolean 預設值是 `false`，上題是 `String` 是類別，不同於基本資料型態，用類別宣告一個變數的意思是宣告一個指向型態為該類別的物件之參考變數

30

```
: 11. String foo = " blue" ;  
12. boolean [] bar = new boolean [1] ;  
13. if (bar[0]) {  
14. foo = "green"  
15. }  
16. System.out.println( " foo = " + foo) ;
```

結果為何？

- A. foo =
- B. foo = blue
- C. foo = null
- D. foo =green
- E. 編譯失敗。
- F. 在執行時期丟出一個例外。

解：第 13 行沒有成立，boolean[0]預設是 false，第 14 行沒有執行

Answer=B

31

```
11. int index = 1 ;  
12. int [] foo =new int [3] ;  
13. int bar =foo [index] ;  
14. int baz = bar +index ;  
15. System.out.println ( "baz =" + baz) ;
```

結果為何？

- A. baz =0
- B. baz =1
- C. baz=2
- D. 編譯失敗
- E. 在執行時前丟出一個例外

解：bar=foo[1]=0，baz=bar+index=0+1=1

Answer= B

32

```
11. public static void main (string [ ] args ) {  
12. int index;  
13. char [ ] array ={ 'x' , 'y' , 'z' };  
14. System.out.println (array [index]);  
15. }
```

結果為何?

- A. X
- B. Y
- C. Z
- D. 編譯失敗
- E. 在執行時前丟出一個例外

解：方法中的區域變數 index 沒有值(不是類別的變數，不會被自動初始化)，沒有值是不能被使用的

Answer= D

33

```
11. public static void main (string [ ] args ) {  
12. int index=-1 ;  
13. char [ ] array ={ 'x' , 'y' , 'z' };  
14. System.out.println (array [index]);  
15. }
```

結果為何?

- A. X
- B. Y
- C. Z
- D. 編譯失敗
- E. 在執行時前丟出一個例外

解：索引必須從 0 開始

_Answer= E

34

```
1. public class test {  
2. private static int [ ] x ;  
3. public static void main (string [ ]args ) {  
4. System.out.println(x[0]);  
5. }  
6. }
```

結果為何？

- A. 0
- B. null
- C. 編譯失敗
- D. 在執行時期丟出一個 `NullPointerException` 例外
- E. 在執行時期丟出一個 `ArrayIndexOutOfBoundsException` 例外

解：x 是一個類別變數，被自動初始化為 null，用 null 存取一個不存在的資料，結果是 D

Answer= D

35

```
1. public class test {  
2. private static int [ ] x =new int [0] ;  
3. public static void main (string [ ]args ) {  
4. System.out.println(x[0]);  
5. }  
6. }
```

結果為何？

- A. 0
- B. null
- C. 編譯失敗
- D. 在執行時期丟出一個 `NullPointerException` 例外
- E. 在執行時期丟出一個 `ArrayIndexOutOfBoundsException` 例外

解：這是一個很奇怪又語法無錯的程式，第 2 行產生一個大小為 0 的整數陣列物件，這個物件確實存在，只是沒有 `x[0]`...

Answer= E

36

```
1. public class test {  
2. private static int [ ] x =new int [1] ;  
3. public static void main (string [ ]args ) {  
4. System.out .println(x[0]);  
5. }  
6. }
```

結果為何?

- A. 0
- B. null
- C. 編譯失敗
- D. 在執行時期丟出一個 nullpointerexception 例外
- E. 在執行時期丟出一個 Arrayindexoutofboundsexception 例外

Answer= A

37

```
1. public class test {  
2. private static float [ ] f =new float [2];  
3. public static void main (string [ ]args ) {  
4. System.out .println( "f[0] =" +f [0]);  
5. }  
6. }
```

結果為何?

- A. f[0]=0
- B. f[0]=0.0
- C. 編譯失敗
- D. 在執行時期丟出一個例外

Answer= B

38

```
1. class alpha {  
2. private static string [ ] names ;  
3.  
4. public static void main (string [ ]args ) {  
5. System.out .println(names[0]);  
6. }  
7. }
```

結果為何？

- A. null
- B. 編譯失敗
- C. 在執行時期丟出一個例外
- D. 程式執行後產生的輸出不確定

解：names 的預設值是 null

Answer= C

39 一個 char 的數值範圍為何？

- A. 0...32767
- B. 0...65535
- C. -256...255
- D. -32768...32767
- E. 範圍因平台不同而有所不同

Answer= B

解：char 是 16 位元的無號數

40 下列哪三項是 float 的有效宣告?(選擇三項)

- A. float foo = -1; -1 預設型態是 32 位元的 int，可
- B. float foo =1.0; 1.0 預設型態是 double
- C. float foo = 42e1; . 同 B
- D. float foo =2.02f; . 可
- E. float foo =3.03d; . 不可
- F. float foo =0x0123; . 可，用 16 進位表示，這邊只列一個 16 位元的值

Answer= ADF

?short 的數值範圍為何?

- A. 0 至 4264967295
- B. 介於 0 至 65536 之間
- C. 介於-32768 至 32767 之間
- D. 介於-2147483648 至 2147483647 之間
- E. 範圍因平台不同而有所不同

解：short 是 16 位元的有號數，和 char 不同

Answer= C

41 下列哪兩項是合法的 String 宣告?(選擇兩項)

- A. String s =null ;
- B. String s = 'null' ; //單引號內只能有單一個字元
- C. String s = (String) 'abc' ;
- D. Stirng s = “this is a sting” ;
- E. String s = “this is a very\nlong string” ; //在 java 原始碼中不可以跳行

Answer= AD

42 下列哪兩項是 float 的有效宣告?(選擇兩項)

- A. float f = 1f ;
- B. float f = 1.0 ;
- C. float f = '1' ; //字元值為 16 位元
- D. float f = “1” ;
- E. float f = 1.0d ;

Answer= AC

43 下列哪兩項是 boolean 的有效宣告?(選擇兩項)

- A. `boolean b = 0 ;`
- B. `boolean b = true ;`
- C. `boolean b = "true" ;`
- D. `boolean b = 'true' ;`
- E. `boolean b = (3>=0) ;`
- F. `boolean b = Boolean.TRUE ;`

Answer= BE

44 一個 byte 的數值範圍為何?

- A. 0...255
- B. 0..65535
- C. -128...127
- D. -32768...32767
- E. 範圍因平台不同而有所不同

Answer= C

45

1. `public class x {`
2. `public static void main (string []args) {`
3. `byte b= 127 ;`
4. `byte c = 136 ;`
5. `byte d = c-b ;`
6. `System .out.println ("d =" +d) ;`
7. `}`
8. `}`

結果為何?

- A. `d =-1`
- B. `d = 255`
- C. 編譯失敗
- D. 在執行時期丟出一個例外

解說：第 4 行，常數 136 超過 byte 所能表示的值，-128~127，因此編譯器會跟你講有問題…

Answer= C

46

11. `int i = 0xFFFFFFFF1;`

12. `int i = ~i;`

13. `System.out.println(j);`

結果為何?

A. 0

B. 1

C. 14

D. -15

E. 編譯失敗

解說：~是 not 的意思，也就是 0 變 1，1 變 0...，將 i 的值以 2 進位的方式來看，看最後面的 16 進值就可以，0001 做 not 後為 1110，就是 8+4+2

Answer= C

47 下列哪兩項是相等的?(選擇兩項)

A. $3/2$

B. $3 < 2$

C. $3 * 4$

D. $3 << 2$

E. $3 * 2^2$

F. $3 <<< 2$

解說：

$3/2 \rightarrow 1.5$ ，因是整數型態，所以去掉小數點為 1

$3 < 2$ 是邏輯比較，得到 false 的布林值

$3 * 4 = 12$

$3 << 2 = 3 * (2^2) = 12$

$3 * 2^2 = 6 \text{ xor } 2 = 0110 \text{ xor } 0010 = 0100 = 4$

$3 <<< 2$ ，沒有 <<< 這樣的運算子，編譯錯誤

Answer = CD

48 下列哪兩項是相等的?(選擇兩項)

- A. $16 > 4$
- B. $16 / 2$
- C. $16 * 4$
- D. $16 >> 2$
- E. $16 / 2^2$
- F. $16 >>> 2$

解：

$$16 > 4 \rightarrow \text{true}$$

$$16 / 2 = 8$$

$$16 * 4 = 64$$

$$16 >> 2 = 16 / (2^2) = 4$$

$$16 / 2^2 = 8 \text{ xor } 2 = 1000 \text{ xor } 0010 = 1010 = 10$$

$$16 >>> 2 = 16 / (2^2) = 16 / 4 = 4$$

$>>>$ 是不帶號右移， $>>$ 是帶號的右移，這意謂一件事，若是正數的情況下，二種運算結果一樣，但是若是負數的話，帶號右移就是會產生負數的結果，而不帶號右移就會將負數變成正的，所謂不帶號右移就是不把最高位元當做符號位元，只是一般的1而已。

Answer = DF

```
11. public static void main( String [] args ){
12. Integer a = new Integer(10);
13. Integer b = new Integer(10);
14. Integer c = a;
15. int d =10;
16. double e = 10.0;
17. }
```

下列哪三項為 true?(選擇三項)

- A. (a == c)
- B. (d == e)
- C. (b == d)
- D. (a == b)
- E. (b ==c)
- F. (d == 10.0)

解：整數和浮點數的比較，只要浮點數去掉小數點後的整數部份，與所有比較的整數一樣即是相等。

a 是型態為 Integer 的物件參考指向第 12 行用 new 產生的 Integer 的物件，b 也是型態為 Integer 的

物件參考指向第 13 行同樣用 new 產生的另一個 Integer 物件，c 也是一個型態為 Integer 的物件參考，

藉由設定和 a 相同的值而指向和 a 參考到的同一個物件，因此 a==c，但是 b 不等於 a 也不等於 c，b 是

指向另一個物件。

Answer = ABF

50

```
1. interface Inter { }
2. class A implements Inter {}
3. class B extends A{
4. B( ) {
5. A[ ] arr = new A[10];
6. boolean b1 = this instanceof Inter;
7. Boolean b2 = arr instanceof Object;
8. System.out.println( "b1 = "+b1+" , b2 = "+b2);
9. }
10. }
```

若建立新實體 B 結果為何？

- A. b1 = true, b2 = true
- B. b1 = true, b2 = false
- C. b1 =false, b2 =true
- D. b1 =false, b2 =false
- E. 編譯失敗

解：類別 A 實作介面 Inter，等同於繼承自 Inter，也就是 Inter 是類別 A 的父類別，類別 B 又繼承類別 A，所以在類別 B 中操作 this(表示型態為 b 的物件)，因此在類別 b 中的 this 是 Inter 和 A 和 B 的 instance，所以第 6 行中的 b1=true 只要用 new 產生出來的東西就是物件！陣列也是，所有物件的最大的爸爸就是 Object(繼承關係)，因此 arr 指向的物件也是 Object 的 instance，b2=true
Answer = A

51

```
1. public class test {
2. public static void main (string []args ) {
3. System.out.println(args.length > 4&&args [4].equals ( "-d" )) ;
4. }
5. }
```

若使用以下命令列啟動程式：

```
java test one two three -d ?
```

結果為何？

- A. true
- B. false
- C. 編譯失敗
- D. 在執行時期丟出一個例外

解：在上面的命令下為 4 個參數，args.length=4，沒有大於 4，得到 false，因此後面沒有必要做，若要做的話，args[4]會造成 D 的 Answer，4 的話，陣列的索引是 0~3。

Answer = C

52

```
1. public class test {
2. public static void main (string args [] ) {
3. System.out.println (6 ^ 3) ;
4. }
5. }
```

結果為何？

- A. 2
- B. 4
- C. 5
- D. 7
- E. 216
- F. 編譯失敗

解：^為 xor 運算，不是 VB 裏面的次方運算

```
0110
```

```
^0011
```

```
-----
```

```
0101=4+1=5
```

Answer = B

53

```
1. public class x {  
2. public static void main (string []args ) {  
3. String s1 =new string ( "true" ) ;  
4. boolean b1 =new Boolean (true) ;  
5. if (s1 .equals (b1) {  
6. System.out.println ( "equal" ) ;  
7. }  
8. }  
9. }
```

結果為何？

- A. Equal
- B. 編譯失敗
- C. 程式執行後未產生輸出
- D. 在執行時期丟出一個例外

解：java.lang.String 這個類別中的 equals() 中要傳入的參數是欲比較的另一個字串物件，若此字串

物件中的字串相等時，傳回值為 true，此例 s1.equals(b1)，其中 b1 不是字串物件，因此傳回 false，

第 5 行的測試不成立，第 6 行不做，所以此題的 Answer 是 C。

Answer = C

54

```
1. class banana {  
2. private int x ;  
3. private static int nth ;  
4. banana ( ) {x =nth ++ ;}  
5. banana (int x ) {this.x = x ;}  
6. public static void main (string [] args ) {  
7. banana b1 =new banana ( ) ;  
8. banana b2 =new banana ( ) ;  
9. banana b3 =new banana ( 1 ) ;  
10. banana b4 =b3 ;  
11. }  
12. }
```

在執行第 7-10 行之後，哪兩項求值為 true? (選擇兩項)

- A. b2==b3
- B. b3==b4
- C. b1.equals (b2)
- D. b2.equals (b3)
- E. b3.equals (b4)

解：equals()方法是用來比較二個物件，這個方法最開始的時候定義在 java.lang.Object 類別中，用來比較二個物件參考到的物件是否相同？也就是說是否指向同一個物件，因此 b3.equals(b4)等效於 b3==b4。

任何一個類別的 equals()方法應該對 java.lang.Object 中的 equals()做 overridden，也就是某個類別的物件比較是要比較什麼？像上題的 String 類別的 equals()方法就是用來比較二個字串物件中的“字串”是否相等，這和在 Object 類別中的 equals()方法比較是否為同一個物件，在行為上是不一樣的。

也就是說，在沒有改寫 equals 方法的情形式下==等同於 equals 方法。

Answer = BE

55

```
11. try {
12. if ( ( new Object ( ) ).equals (( new Object ( ))) ){
13. System.out.println ( “equals” ) ;
14. }else {
15. System.out.println ( “not equal” ) ;
16. }
17. }catch (exception e ) {
18. System.out.println ( “exception” ) ;
19. }
```

結果為何？

- A. equal
- B. not equal
- C. exception
- D. 編譯失敗

解：和上題一樣，類別 Object 的 equals()方法，題目中用二個 new 產生二個物件，此二個物件當然是 不一樣的。

Answer = B

56

```
11. try {
12. if ( ( new Object( ) ).equals (( new String( “x” ) ))) {
13. System.out.println ( “equals” ) ;
14. }else {
15. System.out.println ( “not equal” ) ;
16. }
17. }catch (Exception e ) {
18. System.out.println ( “exception” ) ;
19. }
```

結果為何？

- A. equal
 - B. not equal
 - C. exception
 - D. 編譯失敗
- }

解：new Object()出來的物件其 equals()方法用來比較二個物件是否為同一個。若是 Sting 類別的物件其 equals()就是比較物件內的字串沒有沒一樣，不管是不是同一個物件。

Answer = B

57

10. Object objA =new Object () ;

11. Object objB =new Object () ;

12. Object objC =objA ;

下列哪兩項求值為 true ?(選擇兩項)

A. objA == objB

B. objC == objA

C. objB == objC

D. objA.equals (objC)

E. objB.equals (objA)

F. objC.equals (objB)

Answer = BD

●假定 b 和 c 參考封裝物件類別的實體,下列哪兩項陳述是正確的?(選擇兩項)

A. b.equals (b)傳回 true

B. b.equals (c)傳回與 b == c 相同的結果

C. 即便 c.equals (b)傳回 true , b.equals (c) 仍可傳回 false

D. 假如 b 和 c 屬於不同的封裝物件類型, b.equals (c) 會丟出一個例外

E. 假如被比較的封裝物件類型不同, b.equals (c) 會傳回 false

解：b 和 c 是類別 Object 的實體，b.equals(c)若要傳回 true，前提是要同一個物件，所有的物件實體

皆屬於 Object 類別(最大的爸爸~~~)

Answer = AE

58

11. String s = "hello" +9 +1 ;

12. System.out .println (s) ;

結果為何?

A. Hello91

B. Hello10

C. 編譯失敗

D. 在執行時期丟出一個例外

解：字串的+就是把運算元變成字串後相接，若此題是 9+1+" Hello"，那麼應該是 10Hello，電腦程式

是一行一行執行的…，前面是字串，後面就是字串，+就是字串的相接，前面是數值的話，+就是數值的

相加。

Answer = A

59

20. String s = 1+9+" hello" ;

21. System.out .println (s) ;

結果為何?

A. 19hello

B. 10hello

C. 編譯失敗

D. 在執行時期丟出一個例外

解：看上題的說明

Answer = B

60

```
11. String s = "abcde" ;
12. Object ob = (Object)s ;
13. if (ob.equals(s)) {
14. System.out.println ("AAAA");
15. }else{
16. System.out.println("BBBB") ;
17. }
18. if (s.equals (ob)) ){
19. System.out.println("CCCC") ;
20. }else{
21. System.out.println("DDDD");
22. }
```

下列那兩項為輸出?(選擇兩項)

- A. AAAA
- B. BBBB
- C. CCCC
- D. DDDD

解:s 和 ob 是指向同一個物件，問題在於 ob 指向的 equals() 方法是類別 Object 的 equals() 還是 String 類別中的 equals()，這種問題我們這樣看，類別 String 中的 equals() 方法 overridden 掉類別 Object 的 equals() 方法，也就是型態為 String 的物件當中是看不到類別 Object 的 equals() 方法，因此這邊的 equals() 方法不論透過 ob 和 s 都是使用到類別 String 的 equals() 方法，同一個物件的比較都是 true 的，Answer 就是 AC

Answer= AC

61

```
21. int i = 1 ;
22. int j = i++ ;
23. if ((i == ++j) | (i++ == j )) {
24. i += j ;
25. }
26. System.out.println("i="+i) ;
```

結果為何？

- A. i=1
- B. i=2
- C. i=3
- D. i=4
- E. i=5
- F. 編譯失敗

解：22行等於 `int j=i=1; i++=>2;`，23行的 `if` 中的“`|`”是簡單的 `or` 運算，不是“`||`”，“`|`”二邊都會做，因此 `i=5`，如果“`|`”換成“`||`”，`i` 會是 4，陷阱題！

Answer= E

62

```
21. int i = 1 ;
22. int j = i++ ;
23. if ((i > ++j) && (i++ == j)) {
24. i + = j ;
25. }
26. System.out .println("i = "+i) ;
```

結果為何？

- A. i =1
- B. i =2
- C. i =3
- D. i =4
- E. i =5
- F. 編譯失敗

解：22行，`j=1`，`i=2`，23行，`++j`，`j=2`，`2>2->false`，`&&`後面不必看了，所以 `i=2`

Answer= B

63

- 11. `boolean a = false ;`
- 12. `boolean b = false ;`
- 13. `boolean c = ((a =true) |(b =true)) ;`
- 14. `System.out .println (a+" , "+b+" , " + c) ;`

結果為何?

- A. `true , true , true`
- B. `true , true , false`
- C. `true , false, false`
- D. `false, false, false`
- E. 編譯失敗

解：13 行，`a=true`，`b=true`，`a|b=>true=>c`，Answer 是 A
Answer=A

64

- 11. `int i =0 , j=1;`
- 12. `if ((i++ == 0)|(j++== 2)) {`
- 13. `i = 42 ;`
- 14. `}`
- 15. `System.out.println ("i = " + i + " , j = " + j) ;`

結果為何?

- A. `i =1 , j =2`
- B. `i =1 , j =1`
- C. `i =42 , j =2`
- D. `i =42 , j =1`
- E. 編譯失敗

解：12 行用的是"`|`"，"`|`" 二邊都會做(不像"`||`")
Answer= C

65

```
11. int i =0 , j=1;  
12. if ((i++ == 0)|| (j++== 2)) {  
13. i = 42 :  
14. }  
15. System.out.println ("i = " + i + " , j = " + j ) ;
```

結果為何?

- A. i =1 , j =2
- B. i =1 , j =1
- C. i =42 , j =2
- D. i =42 , j =1
- E. 編譯失敗

解：12行用的是” ||” ，此式 $0==0$ 成立，且 $i++>1$ ，” ||” 後面的比較式不做，所以 $i=42$ ， j 還是 1

Answer= D

66

```
11. int i =0 , j=1;  
12. if ((i++ == 0)& (j++== 2)) {  
13. i = 42;  
14. }  
15. System.out.println ("i = " + i + " , j = " + j ) ;
```

結果為何?

- A. i =1 , j =2
- B. i =1 , j =1
- C. i =42 , j =2
- D. i =42 , j =1
- E. 編譯失敗

解：12行，先比較 $i==0?$ (=true)，後 $i++>1$ ，由於是” &” 不是” &&” ，二邊都要做，結果再” &” ， $j==2?$ (=false)， $j++>2$ ， $true\&false=false$ ，所以 12 行 if 不成立

Answer= A

67

```
11. int i =0 , j=1;
12. if ((i++ == 1)&&(j++== 2)) {
13. i = 42 ;
14. }
15. System.out.println ("i = " + i + " , j = " + j ) ;
```

結果為何?

- A. i =1 , j =2
- B. i =1 , j =1
- C. i =42 , j =2
- D. i =42 , j =1
- E. 編譯失敗

解：和上題不一樣是用” && ”，前面是 false，後面不用做

Answer= B

68

```
1. public class X {
2. public static void main (string []args) {
3. int [] a =new int [1] ;
4. modify (a) ;
5. System.out.println (a[0]) ;
6. }
7. public static void modify (int [] a ) {
8. a [0]++ ;
9. }
10. }
```

結果為何?

- A. 0
- B. 1
- C. 在執行時期丟出一個例外
- D. 由於第 4 行出現錯誤, 編譯失敗
- E. 由於第 4 行出現錯誤, 編譯失敗

解：main.a[]和 modify.a[]都指向第 3 行用 new 建出來的一維整數陣列(其中只有一個元素，且其值被初

始化為 0)，第 8 行對該陣列物件的第 0 個元素++，值是 1，第 5 行存取到的也是同一個資料

(畫圖…)

Answer= B

69

```
1. public class X {
2. private static int a ;
3. public static void main (string []args) {
4. modify (a) ;
5. System.out.println (a) ;
6. }
7. public static void modify (int a ) {
8. a ++ ;
9. }
10. }
```

結果為何?

- A. 0
- B. 1
- C. 編譯失敗
- D. 在執行時期丟出一個例外

解：X.a=0，第4行呼叫 modify 把 X.a 的值給 modify.ae0，++之後變為1，可是外面的 X.a 還是0

Answer="A"

70

```
1. public class X {
2. public static void main (string []args) {
3. String s =new string ("Hello") ;
4. modify (s) ;
5. System.out.println (s) ;
6. }
7. public static void modify (string s ) {
8. s += "world!" ;
9. }
10. }
```

結果為何?

- A. Hello
- B. Hello world!
- C. 編譯失敗
- D. 在執行時期丟出一個例外

解：這個程式建出二個字串物件，" Hello" 和" Hello world" main.s 指向" Hello" ，modify.s 先是指向" Hello" ，後來再指向至新建出" Hello world"
Answer= A

71

```
1. public class Test {  
2. public static void replaceJ (string text ) {  
3. text.replace ( ' j ' , ' l ' ) ;  
4. }  
5. public static void main (string args [] ) {  
6. String text =new string ( "java" );  
7. replaceJ (text) ;  
8. System.out.println (text);  
9. }  
10. }
```

結果為何？

- A. lava
- B. java
- C. 編譯失敗
- D. 在執行時期丟出一個例外

解：字串物件是不可以改變的，類別 String 的 replace()方法並不是要換掉本身字串的字元，而是產生

一個新的字串物件來，若要改變應該要這樣做，text = text.replace (' j ' , ' l '); 將 text 指向新產生的字串物件。

Answer= B

72

```
1. public class Test {  
2. public static void leftShift (int i , int j ) {  
3. i <<= j  
4. }  
5. public static void main (string args [] ){  
6. int i = 4 , j = 2 ;  
7. leftshift (i , j) ;  
8. System.out.println (i) ;  
9. }  
10. }
```

結果為何？

- A. 2
- B. 4
- C. 8
- D. 16
- E. 編譯失敗

解：main.i/main.j 是和 leftShift.i/leftShift.j 是不一樣的，main.i 還是一直是 4

Answer= B

73

```
1. public class Mycircle {  
2. public double radius ;  
3. public double diameter ;  
4. public void setRadius (double radius ) {  
5. this.radius = radius ;  
6. this.diameter = radius * 2 ;  
7. }  
8. public double getRadius () {  
9. return radius ;  
10. }  
11. }
```

下列何項陳述是正確的？

- A. MyCircle 類別已完全封裝
- B. 給定 MyCircle 的 diameter 一定是其 radius 的兩倍
- C. 第 5 和第 6 行應在一個 synchronized 區塊中. 以保證封裝
- D. 可以在不影響其 diameter 的情況下, 設定 MyCircle 物件的 radius

Answer= D

74 下列哪項展示了資料封裝的屬性?(選擇兩項)

- A. 方法為 protected
- B. 實體變數為 private
- C. 實體變數具有預設存取
- D. 實體變數可以直接修改
- E. 唯有透過方法,才能存取和修改實體變數

解：封裝就是屬性完全是 private，透過外界可以存取的方法(public/default)來存取這些封裝起來的變數。

Answer="BE"

75 下列那三項展示了[is-a](繼承)關係?(選擇三項)

- A. public class X {}
public class Y extends X {}
- B. public interface shape {}
public interface rectangle extends shape { }
- C. public interface color { }
public interface shape {private color color ; }
- D. public class species { }
public class animal {private species species ; }
- E. public class person { }
public class employee {
public employee(person person;)}
- F. interface component { }
class container implements component {
private component [] children ;
}

解：繼承可由幾個方式做

1. 使用 extendse 子類別 extends 父類別，或子介面 extends 父介面
2. 使用 implements e 類別 implements 介面

Answer= ABF

76 假定[B is a D]和[B has a C], 下列哪兩項陳述是正確的?(選擇兩項)

- A. D 參考 B
- B. B 參考 D
- C. D 參考 C
- D. B 參考 C
- E. D 繼承自 C
- F. B 繼承自 D

解：has-a 就是一個參考的關係

Answer= DF

77 當應用於實體變數時, 何種修飾符可開啟封裝?

- A. void
- B. final
- C. public
- D. static
- E. private

Answer= C

下列那兩項是對類別實行完全封裝的優點?(選擇兩項)

- A. 類別方法的效能得以改善//效率變差，因為要透過方法間接存取
- B. 類別的實作細節得以隱藏
- C. 類別資料成員的存取修飾符可以省略//省略是 default，不是 private
- D. 使用封裝類別的程式可以直接存取資料成員//封裝屬性是 private，不可以直接存取
- E. 可以在不影響該類別客戶端的情況下, 更改類別的內部作業

Answer= BE

78 下列哪三項陳述是正確的?(選擇三項)

- A. 封裝可促成強固的程式碼
- B. 封裝可促成元件的重複使用
- C. 良好的封裝可隱藏實作資料
- D. 封裝採用的是[is-a](繼承關係模式) //不是，是透過對屬性設成 private，和使用方法來存取
- E. 繼承體系採用的是[has-a](組合)關係模式//is-a

Answer= ABC

79

```
1. class passenger { }
2. class engine { }
3. interface transportvehicle{
4. void loadpassengers ( ) ;
passenger p;
5. }
6. interface helicopter extends transportvehicle {
7. int flyIt (string direction ) ;
8. }
9. abstract class jetstream implements helicopter { }
```

下列何項陳述是正確的？

- A. transportvehicle 具有 passenger
- B. engine 被封裝在 jetstream 類別之中
- C. transportvehicle 介面形成多型行為的基礎
- D. 非抽象類別擴充 Jetstream，可選擇為 loadpassengers () 方法提供一個方法定義

Answer= C

80

```
1. public class ConstOver {
2. public ConstOver (int x , int y , int z ) {
3. }
4. }
```

下列哪兩項重載 ConstOver 建構元？（選擇兩項）

- A. ConstOver() {}
- B. protected int ConstOver () {}
- C. private ConstOver (int z , int y , byte x) {}
- D. public Object ConstOver (int x , int y , int z) {}//override
- E. public void ConstOver (byte x , byte y , byte z) {}

解：考題的“重載”是 overload…，且，

建構子(元)的存取屬性，若沒寫則與類別的存取屬性相同

*建構子不能指定回傳值型態

Answer = AC

81

```
5. public class MethodOver {  
6.     public void setVar (int a , int b , float c) {  
7.     }  
8. }
```

下列哪兩項重載 setVar 方法？（選擇兩項）

- A. private void setVar (int a , float c , int b) {}
- B. protected void setVar (int a , int b , float c) {}//override
- C. public int setVar (int a , float c , int b) {return a;}
- D. public int setVar (int a , int b , float c) {return a;}//override , 但是是錯的
- E. protected void setVar (int a , int b , float c) { return c;} //override , 但是是錯的

解：重載是同名異式，方法名同，參數列必須不同，

Answer = AC

82

```
1. class BaseClass {  
2.     private float x = 1.0f ;  
3.     protected float getVar ( ) { return x ; }  
4. }  
5. class SubClass extends BaseClass {  
6.     private float x = 2.0f ;  
7.     // insert code here  
8. }
```

在插入第七行時，下列哪兩項是方法覆載(override)的有效範例？（選擇兩項）

- A. float getVar () { return x ;}
- B. public float getVar () { return x ;}
- C. public double getVar () { return x ;}
- D. protected float getVar () { return x ;}
- E. public float getVar (float f) { return f ;} //overload

解：覆載的用意在於改變一個方法的行為

覆載必須方法名/回傳值/參數列皆要一樣，存取的屬性必須要愈來愈開放...

Public > protected > default > private

A 的問題是存取屬性為 default，原是 protected，不符合規則

C 覆載的問題在回傳值型態不符

E 是重載不是覆載

Answer = BD

83

1. class BaseClass {
2. private float x = 1.0f ;
3. protected void setVar (float f) { x = f ; }
4. }
5. class SubClass extends BaseClass {
6. private float x = 2.0f ;
7. // insert code here
8. }

在插入第七行時，下列哪兩項是方法覆載的有效範例？（選擇兩項）

- A. void setVar (float f) { x = f ; }
- B. public void setVar (int f) { x = f ; } //overload
- C. public void setVar (float f) { x = f ; }
- D. public double setVar (float f) { return f ; }
- E. public final void setVar (float f) { x = f ; }
- F. protected float setVar () { x = 3.0f ; return 3.0f ; } //overload

解：參考上題的說明

Answer = CE

84

1. public class Outer {
2. public static class StaticInner {
3. }
4. }

下列哪兩項陳述是正確的？（選擇兩項）

- A. 類別 StaticInner 需要靜態初始化區塊。
- B. 類別 StaticInner 需要類別 Outer 的一個實體。
- C. 類別 StaticInner 沒有參考類別 Outer 實體。
- D. 類別 StaticInner 可以存取類別 Outer 的非靜態成員。
- E. 可以使用類別名稱 Outer.StaticInner 來參考類別 StaticInner 的靜態成員。

Answer = CE

85

```
1. package foo ;  
2. public class Outer {  
3.     public static class Inner{  
4.     }  
5. }
```

下列何項陳述是正確的？

- A. 編譯失敗。
- B. Inner 類別的實體可以使用 “ new Outer.Inner() ” 進行建構。
- C. Inner 類別的實體不可在 package foo 之外建構。
- D. Inner 類別的實體僅可在 Outer 類別內部建構。
- E. 在 package foo 的內部，可以使用 “ new Inner() ” 建構 Inner 類別的實體。

Answer = B

86

```
11. interface AnInterface { }  
12. class AnAdapter0 {  
13.     AnAdapter0 ( ) { }  
14. }  
15. class AnAdapter1 {  
16.     AnAdapter1( int i ) { }  
17. }
```

下列哪兩項可建立一個匿名內部類別？(選擇兩項)

- A. AnAdapter1 aa = new AnAdapter1() { } ;
- B. AnAdapter0 aa = new AnAdapter0() { } ;
- C. AnAdapter0 aa = new AnAdapter0(5) { } ;
- D. AnAdapter1 aa = new AnAdapter1(5) { } ;
- E. AnInterface ai = new AnInterface (5) { } ;

Answer = BD

87

```
1. class EnclosingOne {  
2. public class InsideOne { }  
3. }  
4. public class InnerText {  
5. public static void main (String [] args ) {  
6. EnclosingOne eo = new EnclosingOne( ) ;  
7. // insert code here  
8. }  
9. }
```

將下列何項述句插入第 7 行可以建構一個內部類別的實體？

- A. InsideOne ei = eo. New InsideOne() ;
- B. eo. InsideOne ei = eo. new InsideOne() ;
- C. InsideOne ei = EnclosingOne. new InsideOne() ;
- D. EnclosingOne. EnclosingOne ei = eo. new EnclosingOne.() ;

Answer = D

88 下列何項陳述是正確的？

- A. 一個匿名內部類別可宣告為 final 。
- B. 一個匿名內部類別可宣告為 private 。
- C. 一個匿名內部類別可實作多個介面 。
- D. 一個匿名內部類別可存取任何封閉範圍中的 final 變數 。

Answer = D

89 Thread 類別中何種方法被用於建立和啟動新執行緒的執行？

- A. run () ;
- B. start() ;
- C. execute() ;
- D. run (Runnable r) ;
- E. start (Runnable r) ;
- F. execute (Thread t) ;

Answer : B

90

```
1. public class Foo implements Runnable {
2.     public void run ( ) {
3.         system.out.println( "Running" );
4.     }
5.     public void start ( ) {
6.         system.out.println( "Starting" );
7.     }
8.     public static void main ( String[] args ) {
9.         new Thread ( new Foo ( ) ).start( );
10.    }
11. }
```

結果為何？

- A. Running
- B. Starting
- C. 編譯失敗。
- D. 程式執行後未產生輸出。
- E. 在執行時期丟出一個例外。

解：Runnable 介面只有一個方法 `public void run()`，第 5 行的 `start` 方法為 `Thread` 中的方法，此類

別實作 `Runnable` 介面，並非繼承 `Thread`...，若是的話，應是不同的結果了

Answer： A

91 假定某類別擴充 `java.lang.Thread`，您會把執行緒中的執程式碼放在何處？

- A. 建構元
- B. `go ()` 方法
- C. `run ()` 方法
- D. `start ()` 方法

Answer： C

92

```
1. class MyThread extends Thread {  
2. public void run ( ) { System.out.println( "AAA" ); }  
3. public void run ( Runnable r ) { System.out.println( "BBB" ); }  
4.  
5. public static void main ( String[] args ) {  
6. ( new Thread ( new MyThread ( ) ).start( );  
7. }  
8. }
```

結果為何？

- A. AAA
- B. BBB
- C. 編譯失敗。
- D. 程式執行後未產生輸出。

解：必須覆載(overridden) public void run()...

Answer : A

93 何種原因可造成執行緒無法執行？

- A. 結束 synchronized 區塊
- B. 呼叫物件的 wait 方法
- C. 呼叫物件的 notify 方法
- D. 呼叫物件的 notifyAll 方法

Answer : B

94 下列何項陳述是對的？

- A. 若僅有一個執行緒被阻斷在物件的 wait 方法之中，而另一執行緒執行了同一個物件的 notify 方法，則第一個執行緒會立刻恢復執行。
- B. 若一個執行緒被阻斷在物件的 wait 方法之中，而另一執行緒執行了同一個物件的 notify 方法，第一個執行緒仍有可能永遠無法恢復執行。
- C. 若一個執行緒被阻斷在物件的 wait 方法之中，而另一執行緒執行了同一個物件的 notify 方法，則第一個執行緒會恢復執行，這是執行 notify 呼叫的直接和唯一結果。
- D. 若兩個執行緒被阻斷在物件的 wait 方法之中，而另一執行緒執行了同一物件的 notify 方法，則最先執行 wait 呼叫的執行緒會恢復執行，這是執行 notify 呼叫的直接和唯一結果。

Answer : B

95 下列何項陳述是正確的？

- A. 呼叫 wait() 方法，執行緒必須擁有所呼叫物件的鎖。
- B. 呼叫 wait() 方法，執行緒必須擁有所呼叫物件的鎖。
- C. 呼叫 join() 方法，執行緒必須擁有所呼叫物件的鎖。
- D. 呼叫 sleep() 方法，執行緒必須擁有所呼叫物件的鎖。
- E. 呼叫 yield() 方法，執行緒必須擁有所呼叫物件的鎖。

Answer : B

96

- ```
11. public class Alpha {
12. public synchronized void foo () {
13. try {
14. Thread.sleep (1000) ;
15. }catch (InterruptedException e) { }
16. }
17. public synchronized void bar () {
18. }
19. }
```

考慮由一個 Alpha 類型的物件 a 和一個執行緒 b 啟動 a.foo( )。當 b 在第 14 行中處於 sleep( ) 方法時，另一執行緒 c 嘗試啟動 a.bar( )。

下列何項陳述是正確的？

- A. 執行緒 c 成功執行了 a.bar( )。
- B. 執行緒 c 丟出一個 IllegalMonitorStateException。
- C. 執行緒 c 必須先取得物件 a 的鎖，才能繼續執行。
- D. 執行緒 c 擁有此執行緒 a 具有更高的優先權，才能執行 a.bar( )。

解：一個鎖是針對物件上鎖，而不是對一個方法上鎖，也就是說，若有一個執行緒進入了某一物件的同

步區段，沒有其他的執行緒可以進入其他同步化區段

Answer : “C”

97

```
1. class ThreadTest extends Thread {
2. public void run () {
3. while (true) ;
4. }
5. public static void main (String[] argv) {
6. ThreadTest a = new ThreadTest () ;
7. a.start () ;
8. try {
9. a.join () ;
10. }catch (InterruptedException e) { }
11. system.out.println (“done”) ;
12. }
13. }
```

結果為何？

- A. done
- B. 編譯失敗。
- C. 程式執行後未產生輸出。
- D. 在執行時期丟出一個例外。

Answer : “C”

98 下列何項陳述是正確的？

- A. 欲呼叫 join ( ) 方法，執行緒必須擁有目前執行緒的鎖。
- B. 欲呼叫 sleep ( ) 方法，執行緒必須擁有目前執行緒的鎖。
- C. 欲呼叫 yield ( ) 方法，執行緒必須擁有目前執行緒的鎖。
- D. 欲呼叫 notify ( ) 方法，執行緒必須擁有目前執行緒的鎖。
- E. 欲呼叫 notify ( ) 方法，執行緒必須擁有所呼叫物件的鎖。

Answer : E

99.

```
1. public class SyncTest {
2. private int x;
3. private int y;
4. private synchronized void setX(int i) { x = i; }
5. private synchronized void setY(int i) { y = i; }
6. public void setXY(int i) { setX(i); setY(i); }
7. public synchronized boolean check() { return x != y; }
8. }
```

當呼叫來自不同類別時，check 在何種情況下會傳回 true？

- A. check 永遠不會傳回 true。
- B. 當 setXY 被多個執行緒呼叫時，check 會傳回 true。
- C. 當多個執行緒分別呼叫 setX 和 setY 時，check 會傳回 true。
- D. 唯有更改 SyncTest，進而可分別設定 x 和 y 時，check 才會傳回 true。

解：由於同步化的設定，設定 x 和 y 的值會被強迫有個順序，也就是某一個執行緒呼叫 check 方法時，

x 和 y 的值有很大機會被設成不同的值…

Answer： B

100.

在對執行緒中的物件呼叫 wait( ) 之後，下列哪兩種情況可使該執行緒處於可執行狀態？(選擇兩項)

- A. 一個執行緒呼叫該執行緒的 resume( )。
- B. 一個優先權較高的執行緒啟動 yield( )。
- C. 一個優先權較高的執行緒啟動 sleep( )。
- D. 一個執行緒呼叫同一物件的 notify( )。
- E. 一個執行緒呼叫同一物件的 notifyAll( )。

Answer： DE

# 心得報告

(BN95021)許起源

俗話說:[不經一番寒徹骨,焉得梅花撲鼻香],真是一點也不錯,沒有經過一番努力,怎麼能收成美果,當然考證照也是一樣的,雖然很辛苦,但是當領到證書的那刻,卻是非常開心的,當然途中也遇到一些挫折,像遇到許多問題不會解,還有因為工作因素,造成組員間一起研讀時間縮短…等等

但總不能因為這些因素就放棄,我們也想出了一些解決方案,遇到問題就去請教老師,老師也很熱心的怕我們找不到他,留下 SKYPE 聯絡方式,就算在家裡也能跟老師 1 對 1 交談,另外我們還約了一個大家都方便的時間出來聚會,地點像是泡沫紅茶店或是麥當勞速食店,一起唸題庫,如果有問題的地方,也可以馬上進行小組討論,相當方便。

### (BN95022)蔡昀達

既然選擇了考證照一路,就只有考過跟沒考過兩個結果,那當然是沒得選擇一定得過才行,所以往後不管是考證照,或是有什麼重大的考試,一開始的決心很重要,另外也得相信並且與團隊互相扶持,畢竟最後能成功,在於大家的努力與風氣;這中間一定會有挫折與困難,參與證照團隊合作的另一項收穫就是懂得一起危機處理,一起共患難,這才是最誠可貴的部分了。

### (BN95029)張晏誠

考試之前感覺時間非常緩慢,但就在要考的幾天前,緊張的心讓我覺得時間就快到了,問著自己究竟準備好了嗎,我不知道,在還沒看到題目之前我是不會心安的,或許到來的是失望與絕望,但是還好,當我看到題目時,那是希望!

希望的光輝正在閃耀著,我的內心在微笑,非常的激昂,考完之後我笑了,笑的燦爛,笑的王八蛋,因為我過了,在此我要謝謝陪伴我走過的朋友,我愛你們。

(BN95050)鄭旭淳

少玩一點，多看一點，少睡一點，多看一題，不要懈怠，成功就在你  
我右左，

別人玩，我們學，別人學，我們還是學學學，眼前只有一個目標，就  
是證照一定要到手，考試完了，心情非常非常的輕鬆，沒浪費家裡的  
錢，也沒浪費老師每週的教解題目，如然，之前做的讀書計畫，派上  
用場了，每天 100 題的努力，也換回了，美麗的果實，現在隨然考上  
證照了，但路才開始，我還要更努力以赴，別人玩，我們還是要在看  
下去，走上精進的道路上，我要不停的學習在學習，台灣人民加油，  
而我也要加油在加油，最後非常謝謝各位同學、老師、家人的加油與  
鼓勵，才有今天考上 scjp 的我，謝謝大家。

## 附錄

## 光碟資料

專題報告

含:封面、書背、評審、書面資料

模擬試題 (PPT 檔)

模擬試題 (WORD 檔)