

筲箕灣東官立中學  
2013 年度香港中學文憑考試  
資訊與通訊科技  
校本評核

軟件開發  
題目二：保健系統

姓名：李駿杰

班別：6 C

學號：3 0

# 目錄

<b>1 目的與分析</b>	<b>3</b>
1.1 目的	3
1.2 分析	5
1.3 資訊系統平台	6
1.3.1 應用程式	6
1.3.2 網頁	8
1.4 可行性研究	10
1.4.1 技術可行性	10
1.4.2 經濟可行性	10
1.4.3 操作可行性	10
1.5 程序編寫語言	11
1.5.1 第三代語言介紹	12
1.6 總結	13
<b>2 系統設計</b>	<b>14</b>
2.1 硬件平台	14
2.2 系統輸入和輸出	14
2.3 用戶界面	15
2.5 基本結構設計	16
2.5.1 系統流程圖	16
2.5.2 結構圖	17
2.5.3 數據流程圖	18
2.5.4 數據字典	20
2.6 系統實施	21
功能一 登入系統	21
功能二 GOTOXY	21
功能三 主清單	22
功能四：病歷記錄系統（醫生版本專有功能）	23
功能五：醫療機構資訊	24
功能六：健康提示	25
2.7 系統測試	26
2.7.1 單元測試	26

2.7.1.1	登入系統	26
2.7.1.2	主目錄系統	27
2.7.1.3	病歷記錄系統	28
2.7.1.3.1	病歷記錄系統修正	31
2.7.1.4	醫療機構系統	31
2.7.1.5	健康提示系統	34
2.7.1.6	BMI 計算系統	35
<b>3</b>	<b>評估</b>	<b>37</b>
3.1	勵志楊醫生之意見	37
3.2	陳家永先生之意見	37
3.3	曾港超先生之意見	37
3.4	小結	37
<b>4</b>	<b>系統轉換及維護</b>	<b>38</b>
<b>4.1</b>	<b>系統轉換</b>	<b>38</b>
4.1.1	直接切入式轉換	38
4.1.2	並行式轉換	38
4.1.3	階段式轉換	38
4.1.4	引導式轉換	38
<b>4.2</b>	<b>系統維護</b>	<b>40</b>
4.2.1	登入系統	40
4.2.2	防火牆和防毒軟件	40
4.2.3	系統及數據備份	40
4.2.4	系統的可持續性	41
4.2.4.1	收集用戶要求之方法	41
4.2.4.2	監察系統之方法	42
<b>4.3</b>	<b>用戶培訓</b>	<b>42</b>
4.3.1	用戶培訓的方法	42
<b>5</b>	<b>總結與反思</b>	<b>43</b>
5.1	系統功能比本身的構思少	43
5.2	對編寫程序語言 Pascal 有更深刻的印象	43
<b>6</b>	<b>資料來源</b>	<b>43</b>
	<b>附件-醫療資訊系統程式編碼</b>	<b>44</b>

# 1 目的與分析

## 1.1 目的

「香城醫療保健網絡」現打算建立一個資訊系統，向客戶提供服務，滿足他們的需要。我是該機構負責此項目的資訊科技項目經理，我需為此項目提供一個良好的解決方案。

在這個資訊系統中，終端用戶為「香城醫療保健網絡」的客戶，因此這系統必須用客戶們的角度來想，要明白他們對此系統的功能有何需求和期望。

在此需釐清「客戶」的定義，在此設定為一般市民和醫療機構兩方面。

當有人生病需去看醫生，便親自到醫療機構掛號。掛號後，也有機會需等待長時間才能見醫生，對於醫療機構離自己家的病人，他們要付出的時間還要多得很。在人人也講求「快」的香港來說，這是一個很浪費時間的行為。

假若有個供連接互聯網功能的系統，病人可用它來掛號。如果醫療機構在病人家的附近，當快到他見醫生的時候，醫療機構便使用系統來進行通知，這樣病人就可以有更多時間留在家中休息，不需浪費時候到到醫療機構等待，對於病人來說是好事。即使醫療機構不在客戶家的附近，也可先進行掛號，這樣便可以減少病人在醫療機構等待的時間。

如果病人還可以使用此系統輸入有關自己病情的基本資料，比如是否有咳嗽、發燒、頭痛等等，就可以使醫生減少需詢問的問題，等同減少見醫生、留在醫療機構的時間，讓病人可以更快回家休息。

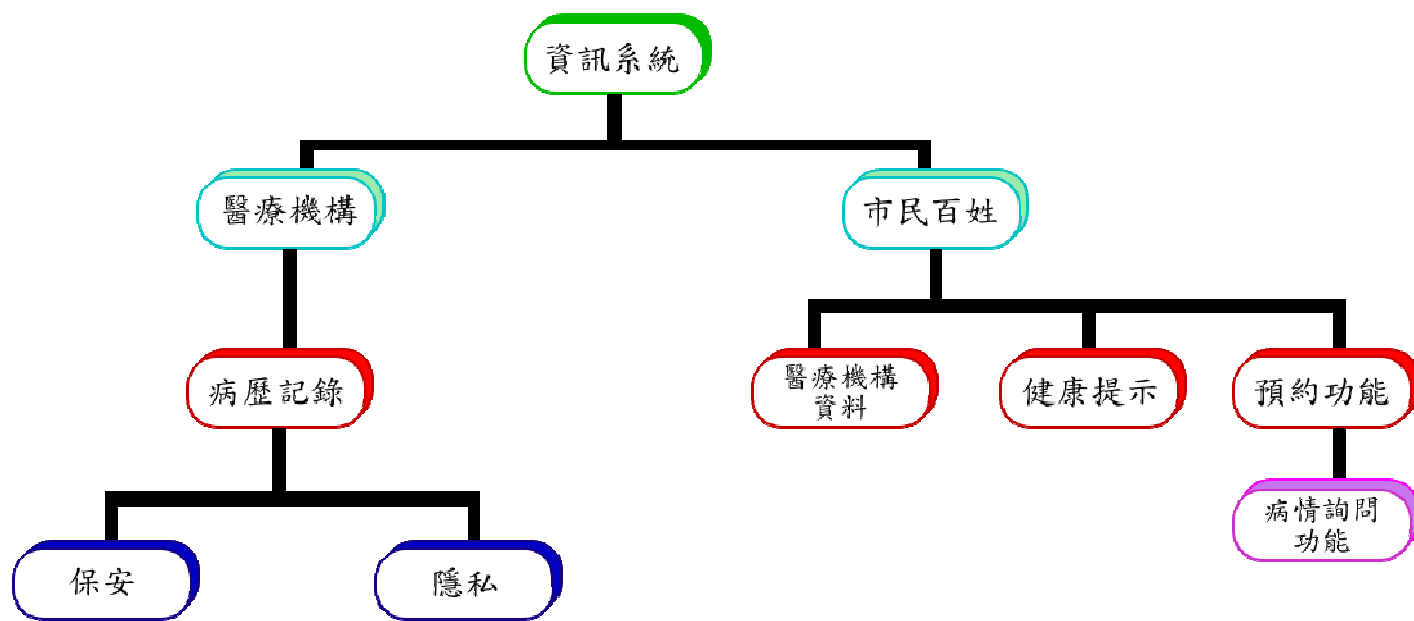
如果系統能提供不同醫療機構的資料(包括地點、診症時間等)，在一些特別日子時客戶就可以知道還有哪些地方有醫務所開放，生病時便可以去這些醫務所看病。

此外，如果可以提供不同病症的成因和處理方法，以及一些健康提示，客戶就更能了解不同的病症和健康常識。

另外，病人的病歷記錄對於醫生來說也是很重要的文件。如果有一個系統可以把病歷記錄下來，把病歷儲存於一個數據庫中，只有使用此系統的醫療機構才能查閱和修改，那麼當病人到使用此系統的醫療機構看病時，醫生就可以對病人的過去的健康有進一步了解。

不過，病歷記錄是個人隱私的一種，因此必須先得到病人的同意，醫療機構才能查閱和修改病人的病歷記錄。

如果在資訊系統中做足以上所指的，相信就能滿足客戶們的需求。不過，為了使客戶可以更快地用慣本公司的資訊系統，系統介面必須是用戶友善，並有使用說明給予客戶，避免他們要用長時間學習使用此資訊系統。



子問題架構

## 1.2 分析

這次的項目是建立一個資訊系統來滿足客戶的需求。 資訊系統的出現，是為了把現行由人手操作的過程自動化，改善現有系統和更正系統問題。

對於現今相關的系統，它們一般也只於醫療機構中，供醫生護士使用，只有病歷記錄功能，並不全面。 因此本次的系統必須改良現今系統功能不全面的問題。

病歷記載是個人私隱的一種。 由於客戶的病歷記載會被上傳至數據庫中，供在醫療機構工作的人員使用，因此的數據庫的保安問題也不容忽視。 在此要設定不同用戶可存取權限，比如只有醫生才可讀取、更改用戶的病歷記載。 在使用前也需輸入密碼，防止被他人更改，保護用戶的私隱。

另外，數據庫的資料必須定時進行備份，確保資訊系統發生故障時，也能迅速地從備份中還原。 此外如果需要更新或升級系統，也必須在更新或升級前進行一次備份，因為在每一次的更新或升級期間也可能發生意外，令系統不能正常使用。

## 1.3 資訊系統平台

在現今社會中，有著不少的資訊系統，它們也是在不同的平台所建立，比如有應用程式或網頁。在建立資訊系統前，必須選擇一種合適的平台，供使用者使用。

### 1.3.1 應用程式



醫療通

醫療通，為香港食物及衛生局所建立的一個具電子平台作用的網上系統，讓醫療券計劃及資助計劃在平台上運作。此系統還能收集病人的主要個人資料，以及管理用於私營基層醫療服務的指定津貼。

由上圖可見，此系統的終端用戶是「服務提供者」和「市民」，系統為這兩類用戶提供不同的服務。

在給「服務提供者」的服務方面，分別有「網上登記」和「已登記服務提供者登入」兩項，讓醫療機構申請使用及管理個人平台。

在給「市民」的服務方面，則有三項。

在「已登記醫療服務提供者名單」中，可以讓市民查詢已提供有關醫療服務的資料，例如服務提供者、執業地址、電話號碼等等。

醫療通－已登記醫療服務提供者名單

「查詢醫療券」中，市民只需輸入個人資料，便可以利用此功能查詢自己所持有的醫療券數目。 同時也提供了網上示範，教授市民如何使用此功能。

最後的「網上預先登記」，則是為方便家長或監護人在兒童前往疫苗注射前，預先填好【使用疫苗資助同意書】所需的資料。 此功能也有提供網上示範，教導市民此功能的用法。

醫療通－網上預先登記



## 1.3.2 網頁



醫療保健

這個網站是把各種疾病的資訊，以及一些健康提示分類，方便瀏覽者搜查有需要的疾病資訊或健康提示。

比如按下「急救常識」，則會顯示下圖的內容。



醫療保健－急救常識（部份）

在「急救常識」版面中按下「10 種常見兒童意外傷害 怎麼處理？」，就會出現有關的資訊。

### **誤食酒精飲料**

如果孩子誤食紅酒，無法確定攝入量，應立即送急診。幼兒喝下少量的酒就會導致酒醉、酒精中毒，甚至嘔吐、窒息和呼吸衰竭。孩童不同於成人，他們的肝臟和肌肉無法儲存糖分，在酒精消退前，血糖值無法保持安全標準，嚴重的低血糖可能導致癲癇發作甚至死亡。最好尋求醫師協助，當孩子酒醒之後，可能需要靜脈注射補充糖分。切記將成人飲料擺放在孩子拿不到的地方。

### **眼睛受傷**

玩具車打傷孩子眼睛，若是一般受傷流血不須送急診，眼周的骨頭能承受撞擊力量，保護眼睛。眼周皮膚鬆弛，只受一點小傷就會瘀青、腫的眼睛張不開，好像很嚴重，其實只要冰敷數天，服用適量止痛藥即可。但若傷及眼球應立即送醫，並查看眼睛是否流血、紅腫、視力模糊或無法轉動、張開眼睛。若眼睛周圍眉骨或眼皮受傷需要手術縫合，先以紗布覆蓋傷口，對傷口直接加壓止血，隨即聯絡兒科醫師並詢問整形醫師。

醫療保健－急救常識－10 種常見兒童意外傷害 怎麼處理？（部份截圖）

## 1.4 可行性研究

進行可行性研究的目的是要測試本項目所建議的方法是否可行。本項目所建議的方法在部分客戶來說，在技術上是可行的，但卻無法負擔成本。因此必須進行可行性研究，分別為技術可行性、經濟可行性、操作可行性。三者同樣重要，在系統設計中必須逐一小心研究。↓

### 1.4.1 技術可行性

對於本項目中的資訊系統中，電腦設備是不可沒有的。因此，如果客戶沒有電腦設備，或電腦需要維修，便不能使用本資訊系統。不過，由於現時大部分家庭都設置了電腦，維修電腦所需的時間相比以往也較短。所以在技術上，這資訊系統是可行的。↓

### 1.4.2 經濟可行性

如客戶想使用本項目中的資訊系統，必須向本公司購買。但如果價格昂貴，不是所有客戶也有能力購買。此外，本資訊系統也必須對客戶帶來好處，否則再好的系統也不會有人購買使用。因為，成本必須定在一個合適的價格上，才会有客戶購買使用。

### 1.4.3 操作可行性

即使本項目中的資訊系統在技術上和經濟上都可行，它也未必能夠切合客戶的需要。不願意接受改變往往是最關鍵的因素。由於本項目的資訊系統需使用電腦，對於上了年紀的人士可能接受不到使用高科技產品，因此放棄使用本資訊系統。此外也有可能是客戶未能應付系統的操作，或不想嘗試使用新的系統，認為現有的教學模式比新的系統好。加上有時在推行新系統時也會產生不少新問題。因此，必須確保新的資訊系統能順利運作，令所有的用戶都接受新系統。↓

## 1.5 程序編寫語言

本項目是建立一個可在電腦上使用的資訊系統，因此必須使用程序編寫語言來規範電腦的工作。 現時的程序編寫語言分為五代，下面將簡單介紹。

第一代語言(1GL)由電腦中央處理器直接執行，由長長的二進制碼所組成。第二代語言(2GL)為把指令以簡寫形式符號化（助記碼），協助程式編寫員記憶指令和寄存器。 此外，這兩代也是機器依賴性的，加上匯編語言程序會在低至寄存器的層面進行電腦記憶體的操作，令學習或除錯上也為困難。

第三代語言(3GL)較接近人類語言，比較像英語，其最大的優點是可以令程序編寫員專注解決目標問題，不必理會繁複的電腦記憶體位址和寄存器，所以使它在編寫、除錯、維護上也比匯編語言較易。 此外，它非機器依賴性的，有較高攜帶性，只需稍作修改，有時完全不必修改，就可以遷移到另一硬件平台使用。

第四代語言(4GL)比第三代語言更抽象化，它所需的語句數目較少，以求節省軟件開發的時間和成本。 第五代語言(5GL)被廣泛應用在人工智能中，程序能自行搜索問題的解決方法。 它摒棄了由程序編寫員設計算法的做法，改由指定的限定和條件解決問題，令程序編寫員可專注在問題本身和所需滿足的限制和條件，不用理會使用什麼算法。

效能 \ 語言	低階語言		高階語言		
	第一代	第二代	第三代	第四代	第五代
攜帶性	低	低	高	高	高
可讀性	最低	低	高	高	高
抽象程度	低	低	一般	高	最高
問題導向	最低	低	一般	高	最高
機器依賴性	有	有	沒有	沒有	沒有
寄存器導向	最低	低	一般	高	最高

表單 1 各代編寫語言的比較

在這五代的編寫語言中，我認為在本項目中該使用第三代語言。 因為它與另外四代的語言相比，它沒有機器依賴性，可讀性高，能在不同的硬件平台使用，抽象度也比第四代和五代語言低。 最重要的一點是，第三代語言為現今常用的編寫語言，對於程序編寫員來說，是較易上手的編寫語言。

## 1.5.1 第三代語言介紹

即使已決定使用第三代語言編寫本項目的資訊系統，但第三代語言也有不少種類。典型的第三代語言包括 FORTRAN、COBOL、BASIC、Pascal、Ada 和 C。其中，Visual Basic、Pascal 和 C 為現今最常用的第三代語言，下面將會簡單介紹它們。

### *Visual Basic*

它是一種簡單易學，容易上手的編寫語言，提供了強大的可視化編程能力，讓程序編寫員輕鬆地做出漂亮的程序，此外眾多的控件讓編寫程序變得簡單。但是，Visual Basic 不是真正面向對象的開發文具，它的數據類型太少，而且不支持指針，使其表達能力很有限，另外只有 Visual Basic 6 或其後的版本編譯出來的代碼才是真正的可執行代碼。

總結來說，Visual Basic 適合初涉編程的人士，因為它對學習者的要求不高，幾乎每個人都可以在一個比較短的時間裡學會此編寫語言。

### *Pascal*

它是一種結構化編程，語言結構嚴謹，可以培養個人的編程思想，讓程序編寫員專有條理地解決目標問題。可惜它有過於嚴謹讓人感覺有點煩，及無法顯示圖像、視像、音樂等非文本內容的問題。

總結來說，Pascal 能使程序編寫員有條理地解決目標問題，但無法顯示圖像、視像、音樂等非文本內容。

### *C*

它是一種靈活性好、效率高、通用的、程序式的程式語言，廣泛用於系統與應用軟體的開發，同時具有高效、靈活、功能豐富、表達力強和較高的移植性等特點，是在程式設計師中備受青睞。只是它對使用者的要求比較高。

總結來說，C 語言是在程式設計師中備受青睞，它能編寫出來的程式功能較另外兩種語言多，是常用的第三代語言。

## 1.6 總結

經過上述的分析和建議，我認為本項目的資訊系統除了是一個應用程式，讓用戶安裝在電腦上，還要能有連接互聯網之功能，讓用戶使用其功能和方便進行系統更新。另外還要提供使用說明和用戶友善，方便用戶使用。此外，數據庫的資料必須定時進行備份，確保資訊系統發生故障時，也能迅速地從備份中還原。

接著對於此資訊系統提供給予客戶的功能，可能分為以四種：病歷記錄、健康提示、預約功能、醫療機構資料。這四種功能分別告訴醫療機構和屬於普通市民的客戶使用。

最後，我建議使用第三代語言(3GL)的 C 來進行編寫，因為它所編寫出來的程式功能較另外兩種語言多，是一種廣泛用於系統與應用軟體的開發。

但由於我只需提供解供方案，因此我只會利用 Pascal 來進行系統中部份功能的概念，作為日後程序編寫員編寫程序時的一個參考。



## 2 系統設計

為了提供問題的解決方案，把用戶需求轉為處理和方法，我將以幾個範疇介紹醫療資訊系統的設計原理和方法，進行系統設計。

### 2.1 硬件平台

一般來說，使用電腦作為本系統的硬件平台，我認為是不二之選，因為它的靈活度極高，能處理不同類型的問題。不過電腦普遍也能分為個人電腦和大型電腦兩種。

大型電腦是以高計算能力和高可靠性見稱，專為處理大量外部數據運計而設，特點是可靠、隨時可用和提供優質服務，因此被廣泛用於大型企業的工資系統中。

相反，個人電腦的計算能力和存貯容量較大型電腦欠佳，因此只適合用於小型系統上。

由於這次的醫療資訊系統要有處理大量外部數據運計、存貯大量數據等等因素，因此我建議本公司選擇使用大型電腦。

### 2.2 系統輸入和輸出

醫療資訊系統的輸入包括了資金、人力、資訊、硬件等。例如公司需要使用資金來購買所需硬件和招聘程式編寫員編寫程式。此外系統亦需要資訊科技技術員來維護系統的運作。

在輸出方面，要包括資訊、產品和服務。系統必須能給予用家合適的資訊服務，否則系統將不會得到用家的使用。

## 2.3 用戶界面

一個好的用戶界面，決定了一個系統的可用性，即用戶是否容易學習和使用該系統。一般來說，用戶界面可分成命令行界面（CLI）和圖形用戶界面（GUI）兩種。

命令行界面（CLI）是指用戶需透過鍵盤輸入各種命令，以控制系統，一般應用於 MS-DOS 或 UNIX 等操作系統。

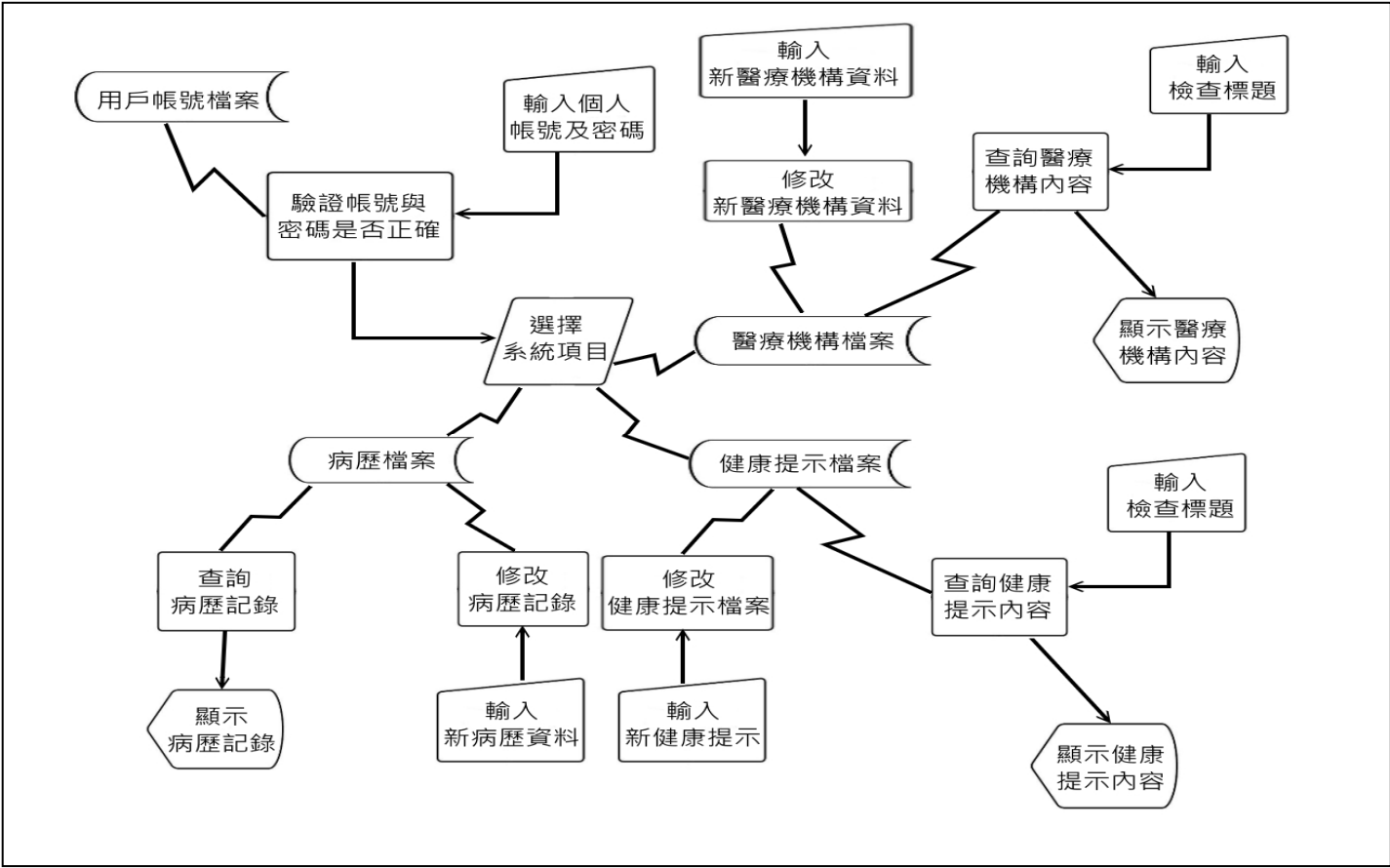
圖形用戶界面（GUI）則讓用戶利用滑鼠或觸式屏幕，選取選單項目或點擊按鈕來輸入命令，能促進用戶和設備之間的溝通。較新式的操作系統，例如 Microsoft Windows 和 Mac OS X 便是採用了圖形用戶界面。

對於本公司的醫療資訊系統，我建議採用圖形用戶界面（GUI）。因為它跟命令行界面（CLI）不同，不需要用戶學習和熟記命令的語法規則，方便用戶使用。



# 2.5 基本結構設計

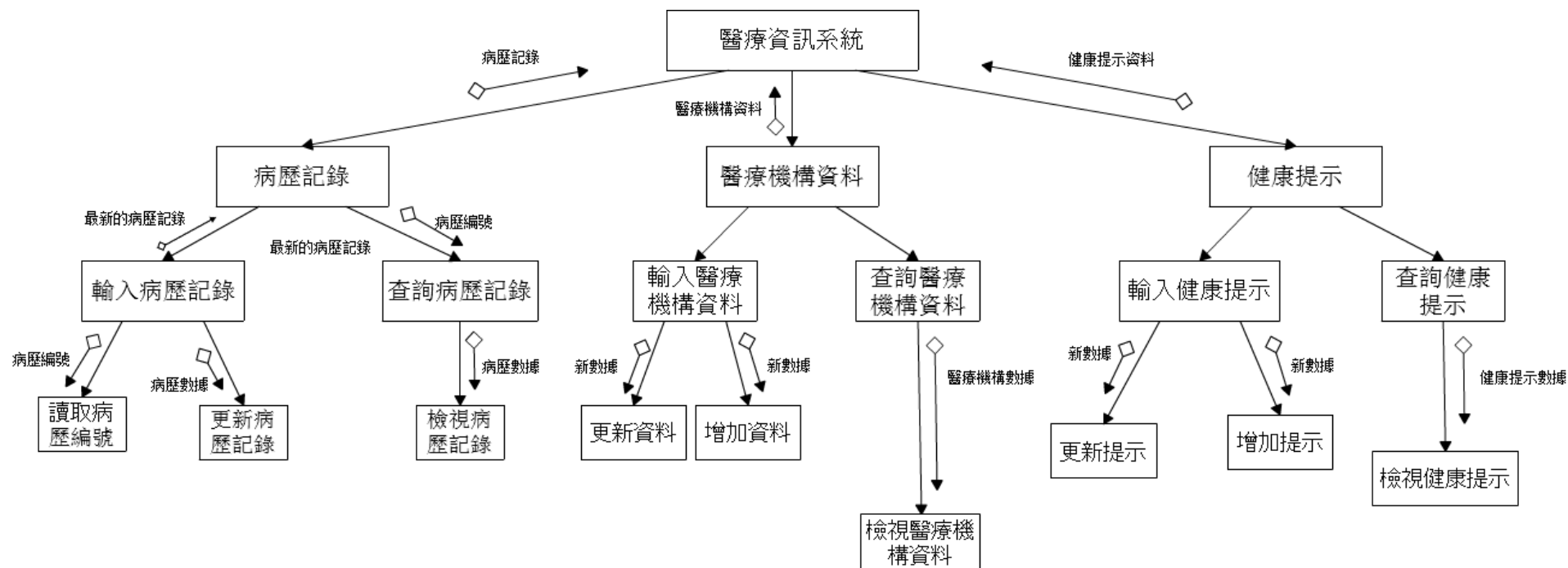
為了能表達系統處理的方法，我將以系統流程圖、結構圖、數據流程圖及數據字典作為解說。



## 2.5.1 系統流程圖

在此系統流程圖中，清楚展示了系統中的各個處理、檔案、數據庫及相關的人手過程。

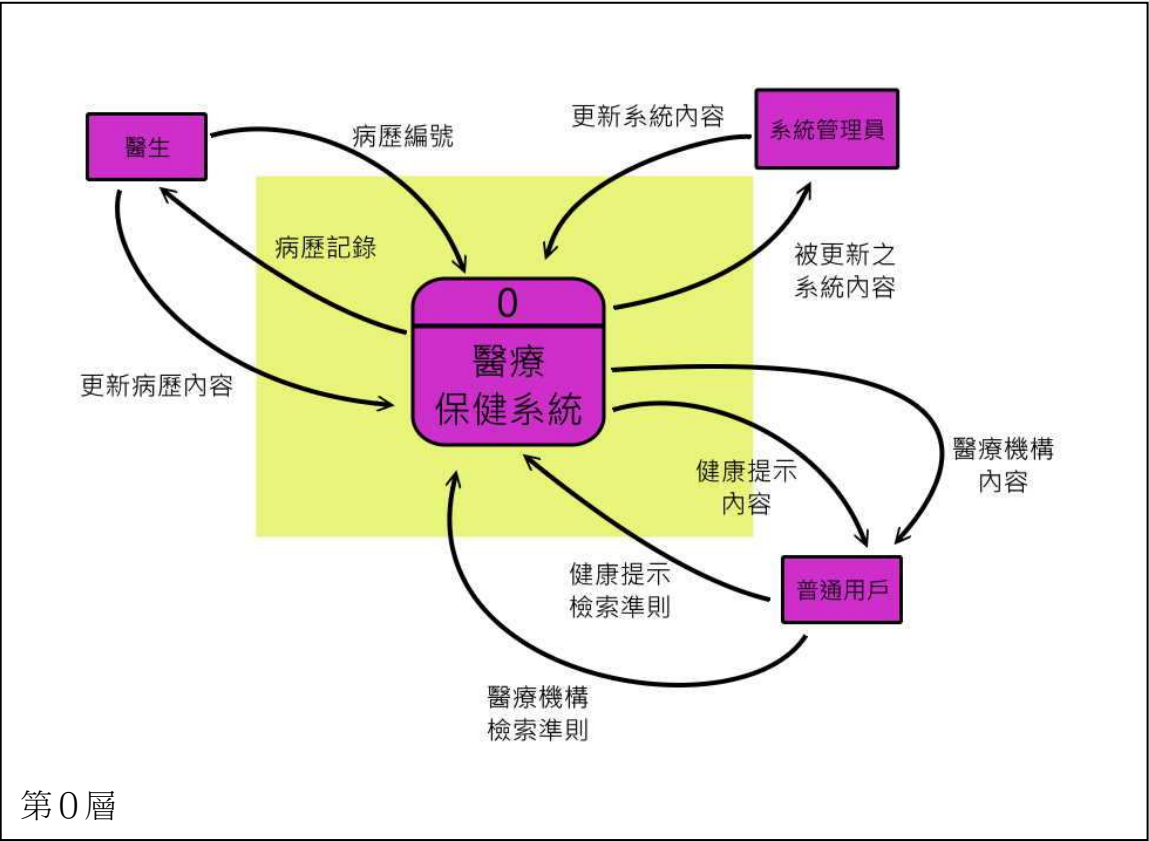
## 2.5.2 結構圖



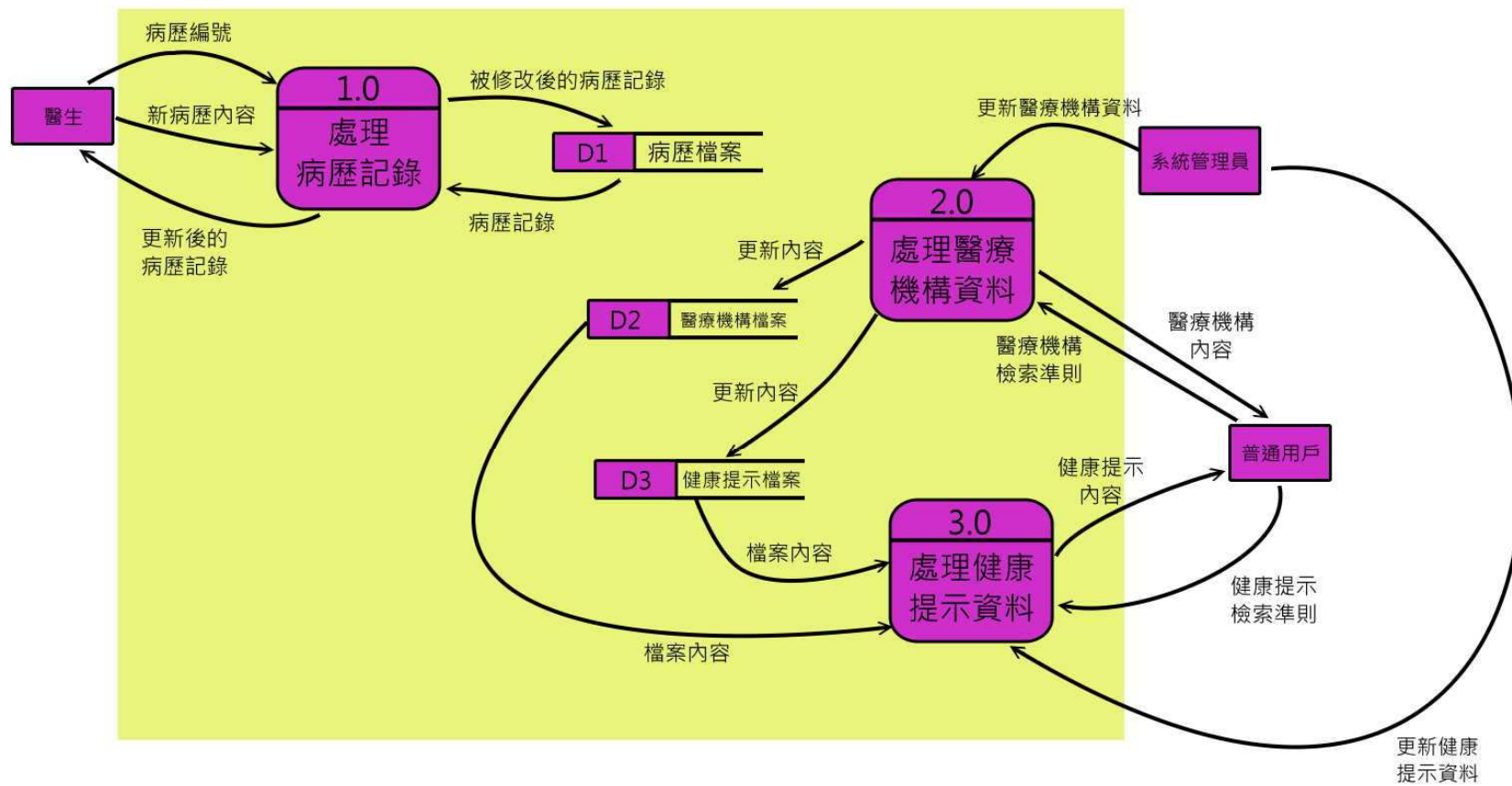
在這張結構圖中，可見系統中不同模組之間的關係。

它把建議的系統功能由上而下逐層分解，描述了獨立模組之間的互動，以及它們之間所傳送的數據。

2.5.3 數據流程圖



數據流程圖顯示了系統所有的需求： 各個處理，進入、離開系統及在系統內的數據或資訊，以及系統貯存資訊的方式。



第一層

## 2.5.4 數據字典

對於數據流程圖中所描述的數據，數據字典能有系統地組織及記錄。

系統用戶 = 帳號 + 密碼 + 系統用戶類別

系統用戶類別 = [醫生 / 普通用戶 / 系統管理員]

健康提示資訊 = 項目名稱 + 內容

醫療機構資料 = 名稱 + 地址 + 電話 + 傳真號碼 + 電郵地址 + 官方網址

病歷記錄清單 = 病歷編號 + 病人姓名

病人基本資料 = 病人記錄編號 + 家屬資料 + { 診証日期 + 病情 + 處方藥物  
+ 是否給予請假紙 }

病人基本資料 = 病歷編號 + 姓名 + 性別 + 身份証號碼 + 出生日期

病人聯絡資料 = 病歷編號 + 住址 + 固網電話 + 手機號碼 + 住址 + 地區

家屬資料 = 姓名 + 性別 + 與病人的關係 + 固網電話 + 手機號碼 + 住址  
+ 地區

病歷記錄 = 診証日期 + 診証原因 + 處方藥物 + 是否給予病假紙

診証日期 = 日 + 月 + 年

性別 = [男 / 女]

住址 = 地方 + 地區

## 2.6 系統實施

以下，將簡單介紹本醫療資訊系統的功能。

### 功能一 登入系統

```
Account:vert_lu  
Password:678901  
Login Successful.  
Press any key to continue...
```

登入成功後的畫面


```
Account:sen  
Password:678901  
Wrong Account or Password.  
Press any key to continue...
```

登入失敗後的畫面

### 功能二 GOTOXY

GOTOXY 會在運行另一功能時，把前一功能留下的內容刪去，重新開始。

```
-----Main List-----  
1.Medical Institutions.  
2.Health Tips.  
3.Exit.  
Please choose the function:
```



```
-----Main List-----  
1.Case Files.  
2.Medical Institutions.  
3.Health Tips.  
4.Exit.  
Please choose the function:
```

## 功能三 主清單

```
-----Main List-----  
1.Case Files.  
2.Medical Institutions.  
3.Health Tips.  
4.Exit.  
Please choose the function:
```

(醫生版本)

1. 病歷記錄系統
2. 醫療機構資訊
3. 健康提示
4. 離開

```
-----Main List-----  
1.Medical Institutions.  
2.Health Tips.  
3.Exit.  
Please choose the function:
```

(普通用戶版本)

1. 醫療機構資訊
2. 健康提示
3. 離開

```
719 procedure ch_funtn(index:integer); {主目錄系統}  
720 begin  
721     goto_XY;  
722     with AC_PW[index] do  
723         if CLSS = 'doctor'  
724         then begin  
725             writeln('-----Main List-----');  
726             writeln('1.Case Files.');727             writeln('2.Medical Institutions.');728             writeln('3.Health Tips.');729             writeln('4.Exit.');730             write('Please choose the function: ');  
731             readln(ch_no);  
732             case ch_no of  
733                 1:CFs;  
734                 2:MIIs;  
735                 3:HTs;  
736                 4:ch_no:=10;  
737                 else writeln('error!');738             end;  
739             goto_XY;  
740         end  
741         else begin  
742             writeln('-----Main List-----');  
743             writeln('1.Medical Institutions.');744             writeln('2.Health Tips.');745             writeln('3.Exit.');746             write('Please choose the function: ');  
747             readln(ch_no);  
748             case ch_no of  
749                 1:MIIs;  
750                 2:HTs;  
751                 else ch_no:=10;  
752             end;  
753             goto_XY;  
754         end;  
755     end;
```

## 功能四：病歷記錄系統（醫生版本專有功能）

```
-----Case Files-----
1.Look at the case file.
2.Exit
Please choose the function:
```

1. 查閱病歷記錄
2. 離開

```
-----Look at the case file-----
Please enter the Case File no. :
```

若選擇「查閱病歷記錄」，使用出現一個此介面，系統將要求用家輸入病歷記錄編號。

```
Case File no. :001
Name :LI CHUN FUNG

1.Personal Profile
2.Communication Information
3.Family Information
4.Sick Information
5.Exit.
Please choose the function:
```

輸入病歷記錄編號後，  
便會出現此介面。

病歷記錄編號：

病人姓名：

1. 個人資料
2. 聯絡方法
3. 家人資料
4. 病歷資料
5. 離開

```
-----Personal Profile-----
Case File no. :001
Name :LI CHUN FUNG
Sex: M
HKID :Y1234567
Day of birthday :8/9/1996

Press any key to continue...
```

選擇「個人資料」，便會  
讀取相關內容，及顯示於  
介面之上。

病歷記錄編號：

病人姓名：

病人性別：

身心號碼：

出生日期：

「查閱病歷記錄」中的另外有個功能，與「個人資料」的運作方式一樣，只是讀取內容不同。



## 功能五：醫療機構資訊

```
-----Medical Institutions-----  
1.Public Hospitals And Institutions  
2.Private Hospitals  
3.Exit.  
Your choose is:
```

1. 公共醫院  
及醫療機構
2. 私營醫院

```
1.Pamela Youde Nethersole Eastern Hospital  
2.Tai Po Hospital  
3.Exit.  
Your choose is:
```

若選擇「公共醫院及醫療機構」，便會出現有關公共醫院及醫療機構的名稱。  
(由於本次作一測試，因此資料中只有兩間醫院的資料)

輸入編號後，便會出現有關該醫院或醫療機構的資訊，包括：  
名稱；地址；電話；傳真號碼；電郵地址；官方網址

```
Name: Tai Po Hospital  
Add.: 9 Chuen On Road, Tai Po, NT  
Tel.: 2607 611  
Fax.: 2665 003  
E-Mail: tph_gr@ha.org.hk  
Web Add.: http://www3.ha.org.hk/tph  
  
Press any key to continue...
```

## 功能六：健康提示

```

-----Health Tips-----
1.Hair Loss
2.Male Diet
3.Male Sports
4.Men's Health
5.Women's Health
6.Check BMI
7.Exit.
Your choose is:

```

身高體重指數(BMI)計算器

選擇使用此功能，便會出現此介面。

用戶需選擇功能，首五項為與保健有關的資訊，第六項為「身高體重指數(BMI)計算器」。

```

-----Check BMI function-----
Enter you height(m):1.75
Enter you weight(kg):61
Your BMI is: 19.92
Your weight status is pass!

Press any key to continue...

```

用戶只需輸入身高及體重，系統將會自動計算出結果。

```

508 procedure checkBMI; {BMI計算系統}
509 var
510     height,weight,bmi:real;
511 begin
512     goto_XY;
513     writeln('-----Check BMI function-----');
514     write('Enter you height(m):');
515     readln(height);
516     write('Enter you weight(kg):');
517     readln(weight);
518     bmi:= weight/(sqr(height));
519     writeln('Your BMI is: ',bmi:1:2);
520     if BMI<18.5 then
521         writeln('Your weight status is underweight!')
522     else
523         if (BMI>24.00) and (BMI<28.00) then
524             writeln('Your weight status is overweight!')
525         else
526             if BMI>28.00 then
527                 writeln('Your weight status is obese!')
528             else
529                 writeln('Your weight status is pass!');
530     writeln(' ');
531     writeln('Press any key to continue...');
532     readln();
533     goto_XY;
534 end;

```

## 2.7 系統測試

為確保系統可以如期運作，我將進行系統測試。

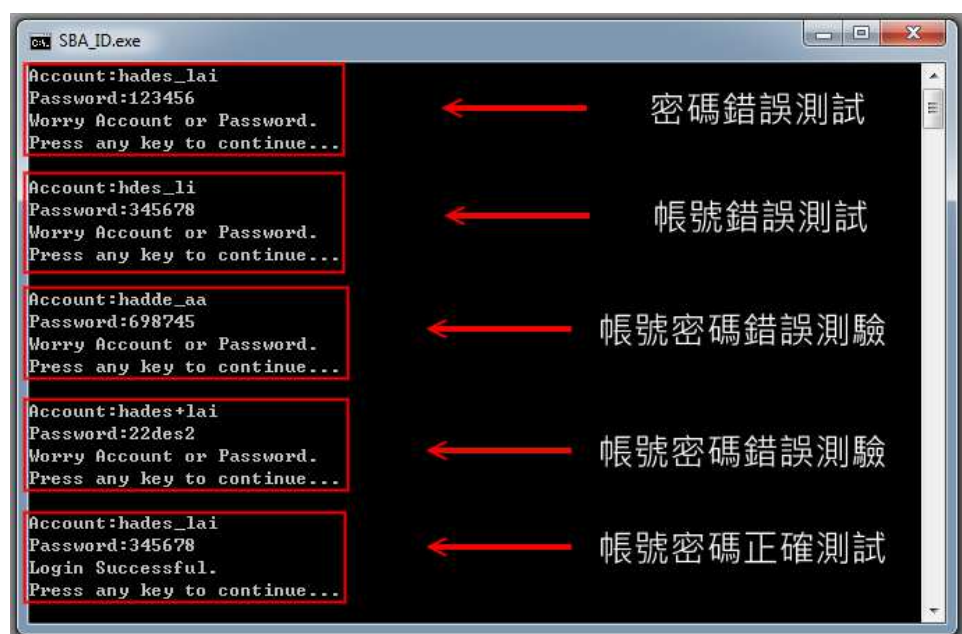
系統測試分為三種：由程序編寫員進行的「單元測試」；由測試團隊進行的「系統測試」及由用戶進行的「驗收測試」。

在此，我將只會進行「單元測試」，日後發布此醫療資料系統前再交由測試團隊進行「系統測試」，測試系統的容量、存貯、性能時間等問題。發布後亦需請用戶進行「驗收測試」，收集他們對系統的滿意程度。

### 2.7.1 單元測試

以下我將會把系統分解成多個不同的模組，再使用「單元測試」，找出各模組的錯誤或故障問題，待日後把所有模組合併成一個大系統時，減低錯誤或故障問題的出現機會。

#### 2.7.1.1 登入系統



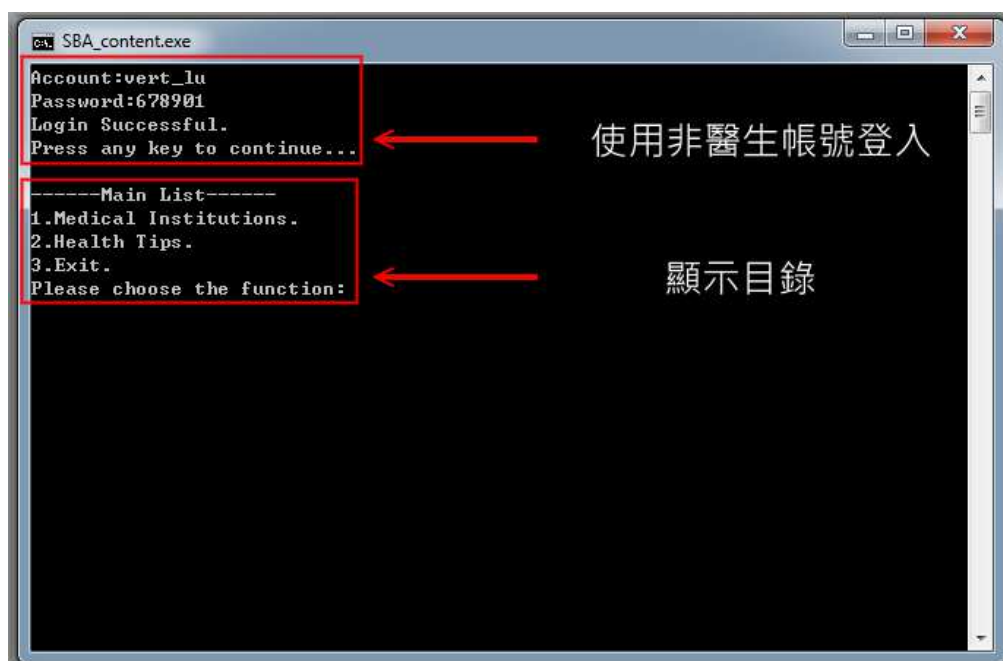
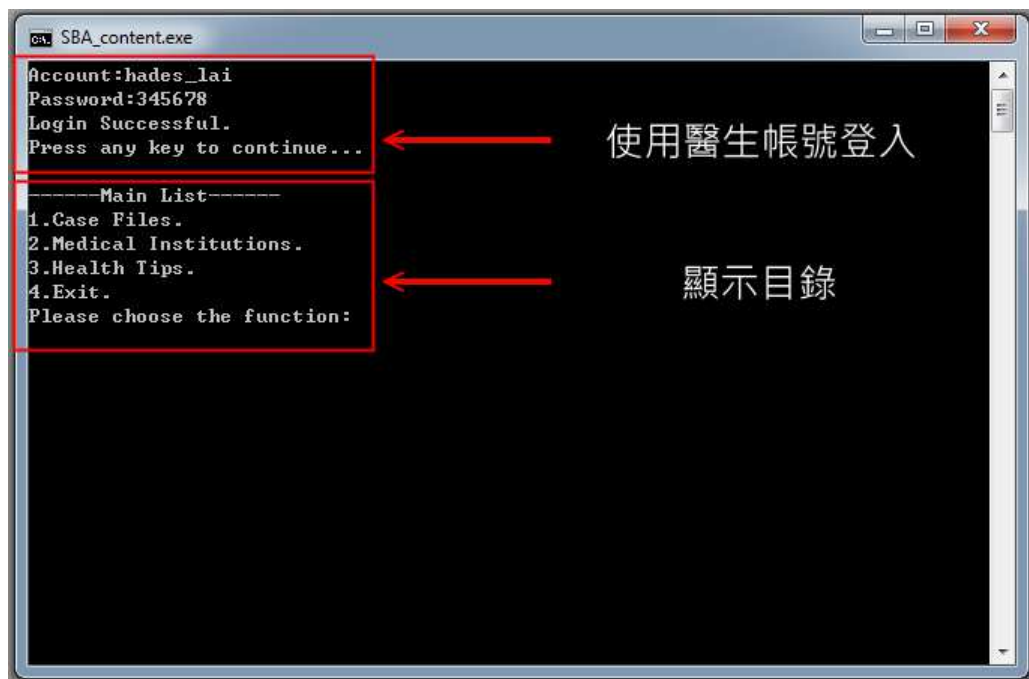
經過使用不同組合的數據輸入測試後，系統亦正常運行，但卻發現誤把 wrong 打錯成 worry。

```
135 then begin
136     writeln('Wrong Account or Password.');
```

```
137     writeln('Press any key to continue...');
138     ch_no:=7;
```

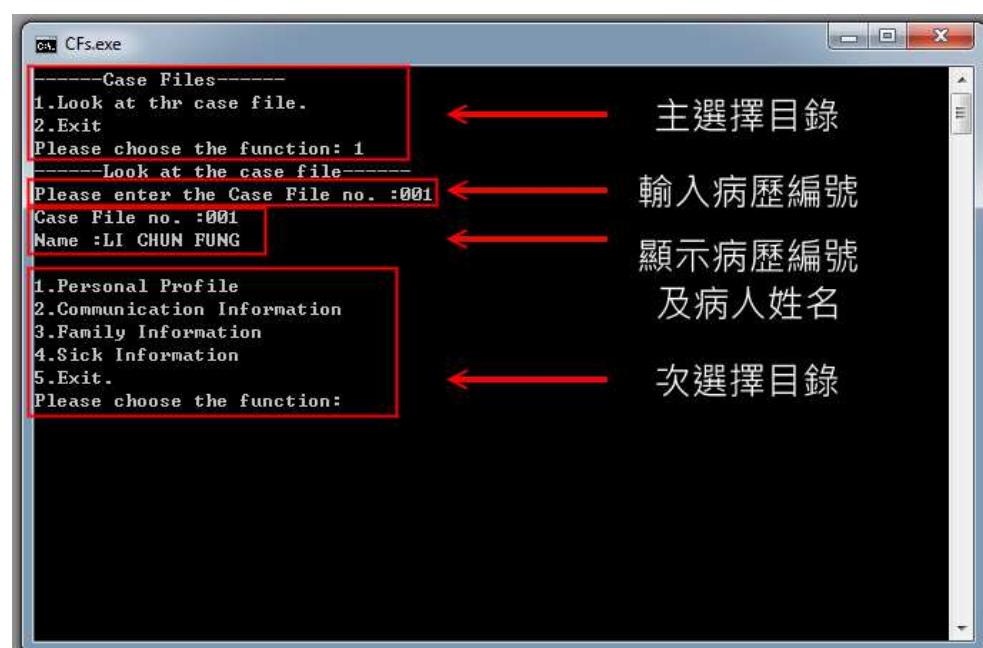
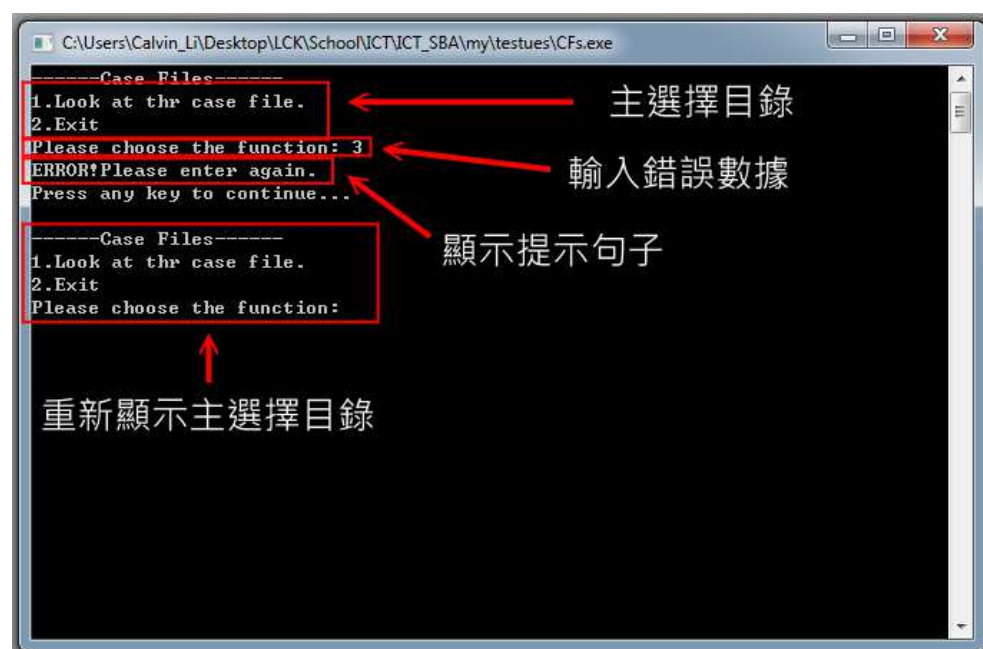
經修改後，便沒出現串錯字的問題。

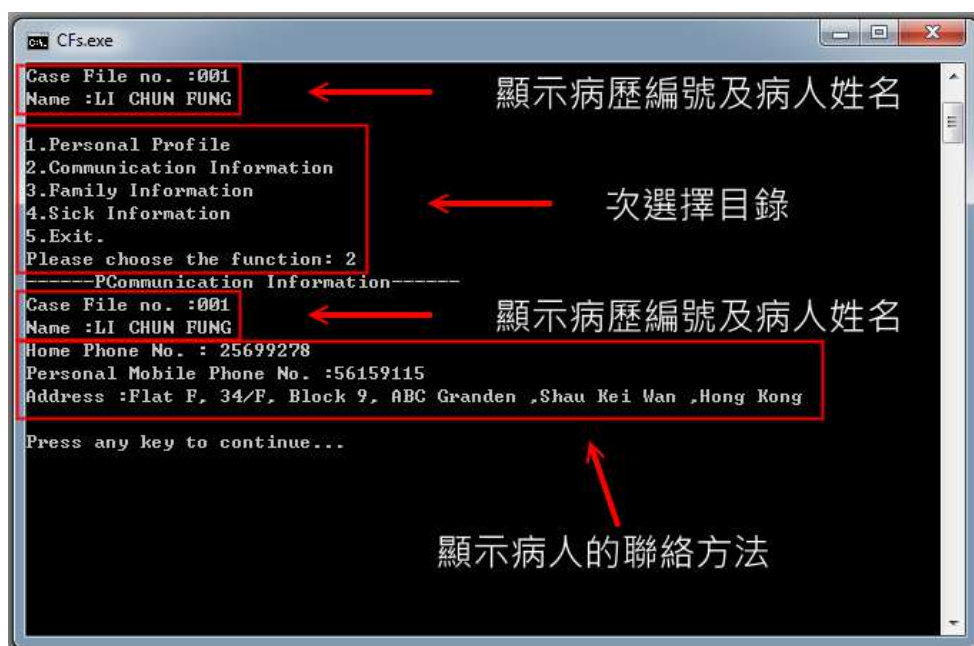
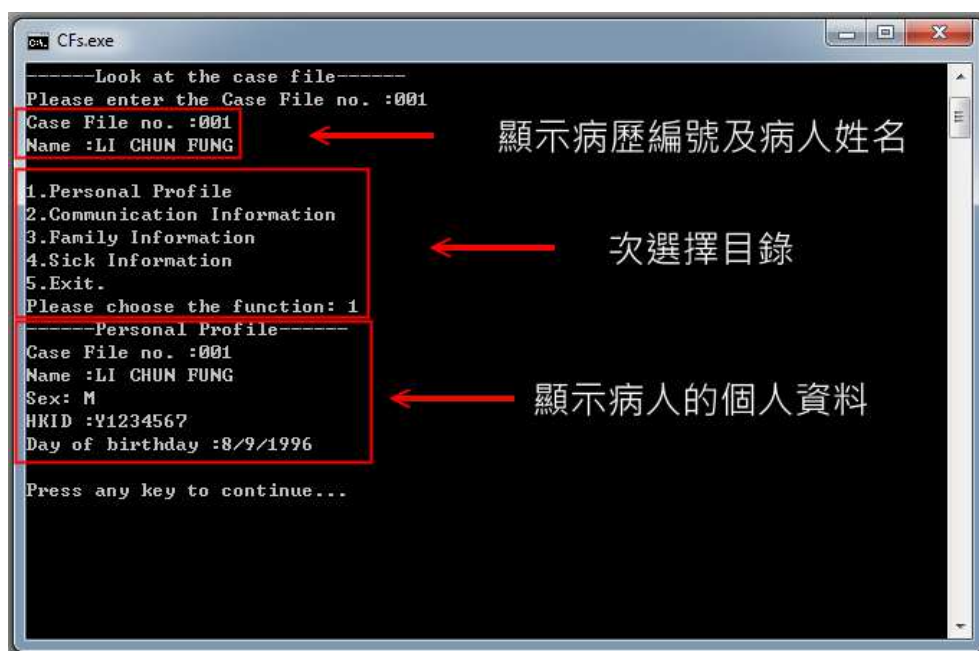
### 2.7.1.2 主目錄系統

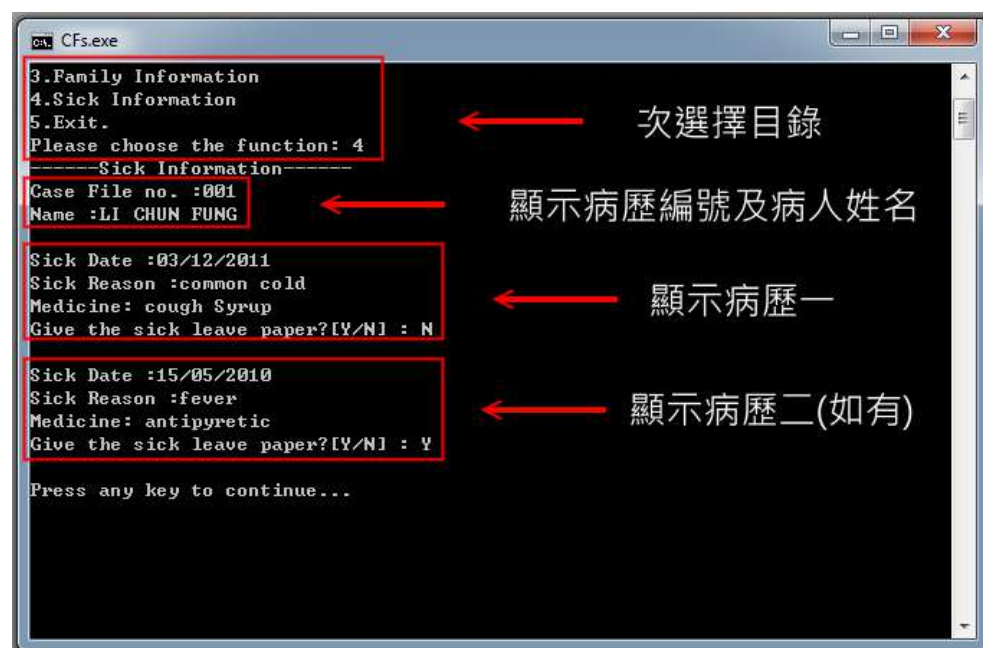


經過使用不同兩組不同類別的帳號（醫生帳號／非醫生帳號）測試後，系統亦能顯示出正確功能的主目錄，表示主系目錄統並無問題。

### 2.7.1.3 病歷記錄系統







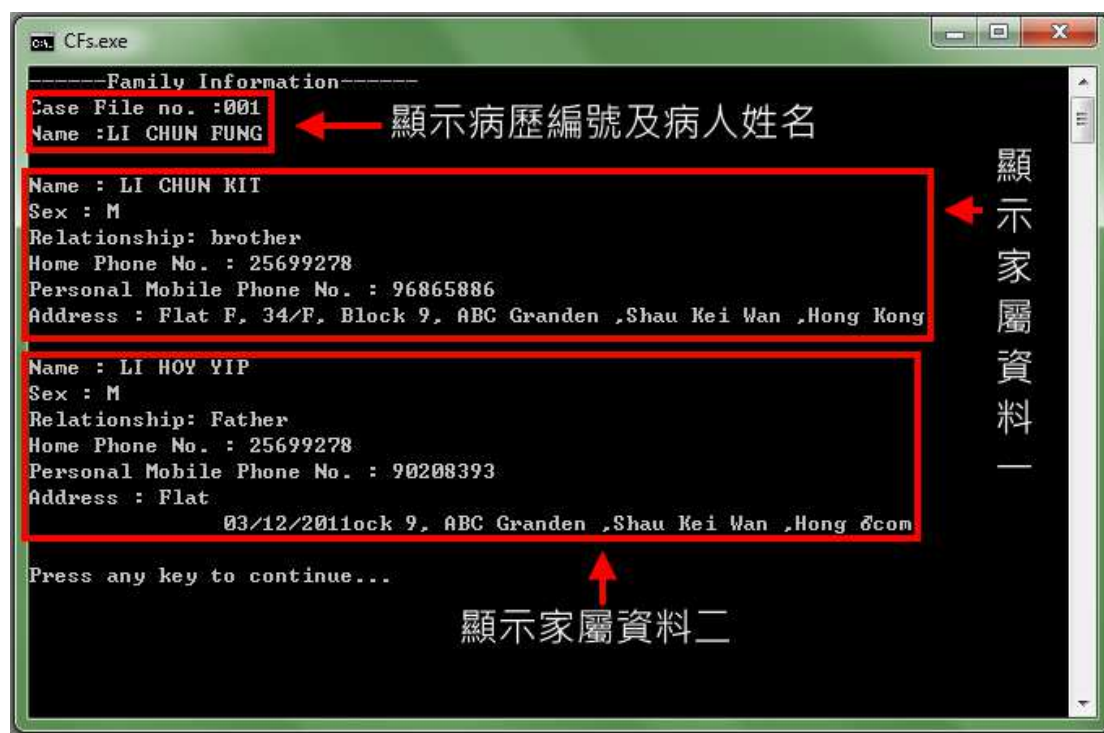
經過測試後，發現把 the 誤串成 thr，另外可發現功能 'Family Information' 出現錯誤，不能正確顯示資料，需要修正。此外，在功能 'Communication Information' 中顯示為 'PCommunication Information'，需要修正。其他功能，在測試後，並沒發現錯誤。

```
454 CFsListDataInput;
455 writeln('-----Look at the case file-----');
456 write('Please enter the Case File no. :');
```

經修改後，便沒出現串錯字的問題。



### 2.7.1.3.1 病歷記錄系統修正

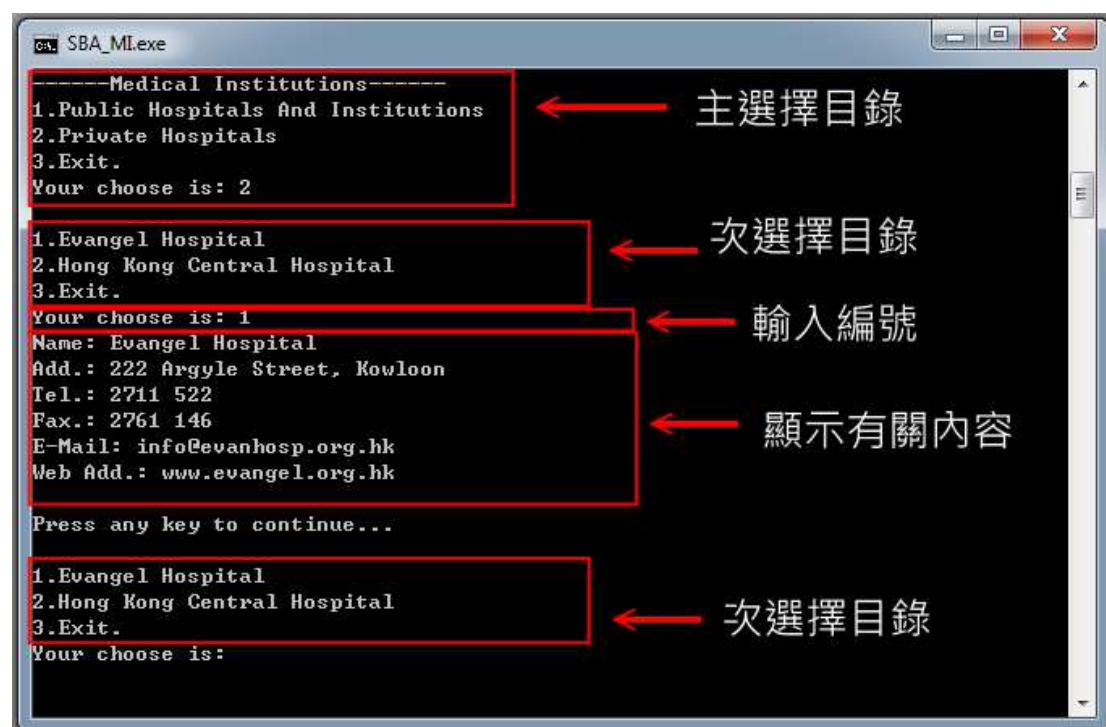


在功能 'Family Information', 發現並沒設定家屬資料的數目, 令系統運行時出現邏輯錯誤。修正後的版本已成功顯示出家屬資料二, 可是在 'Address' 中出現了小問題, 需要後期再進行修正。

### 2.7.1.4 醫療機構系統



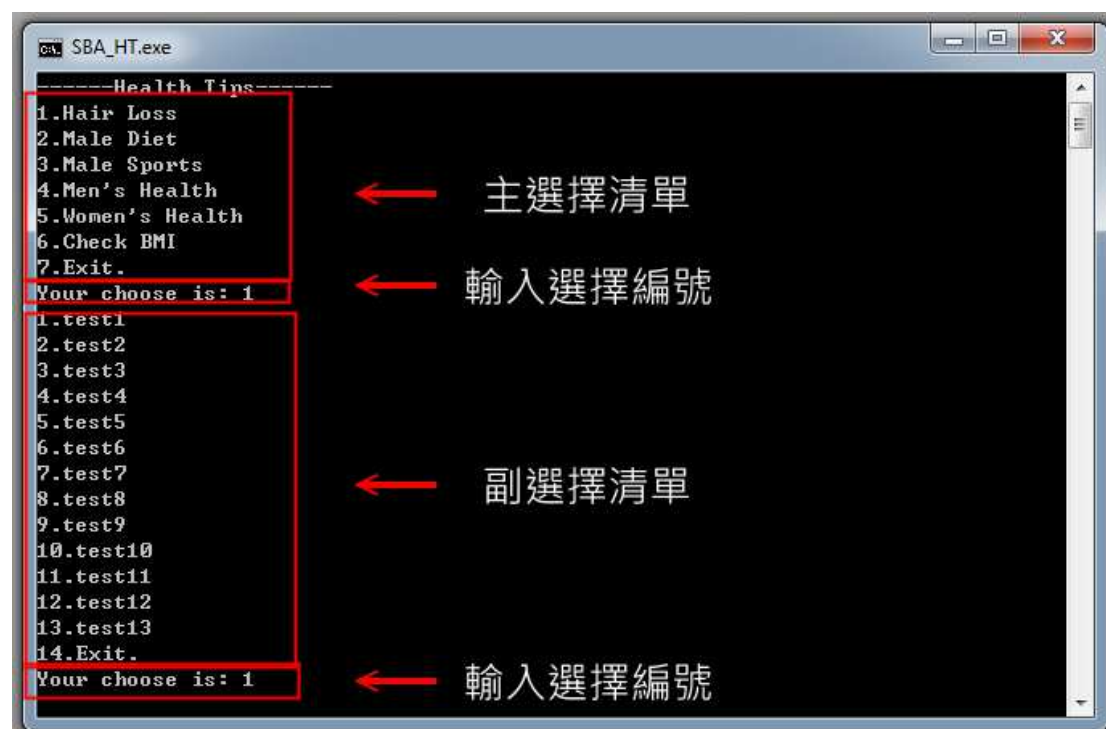






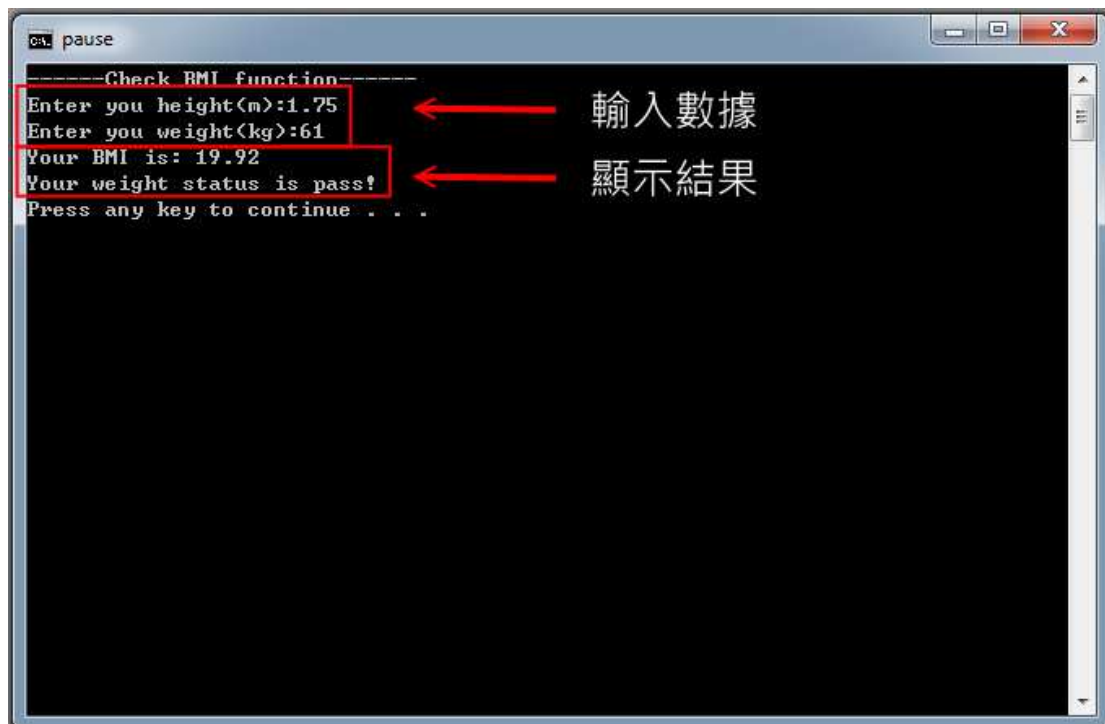
經過測試後，可見系統能運作正常，並無錯誤。

## 2.7.1.5 健康提示系統



經過測試後，可見能運作正常，並無錯誤。由於功能的程序與已測試功能的程序一樣，因此不作測試。另外，功能 'Check BMI' 見「2.7.1.6 BMI 計算系統」。

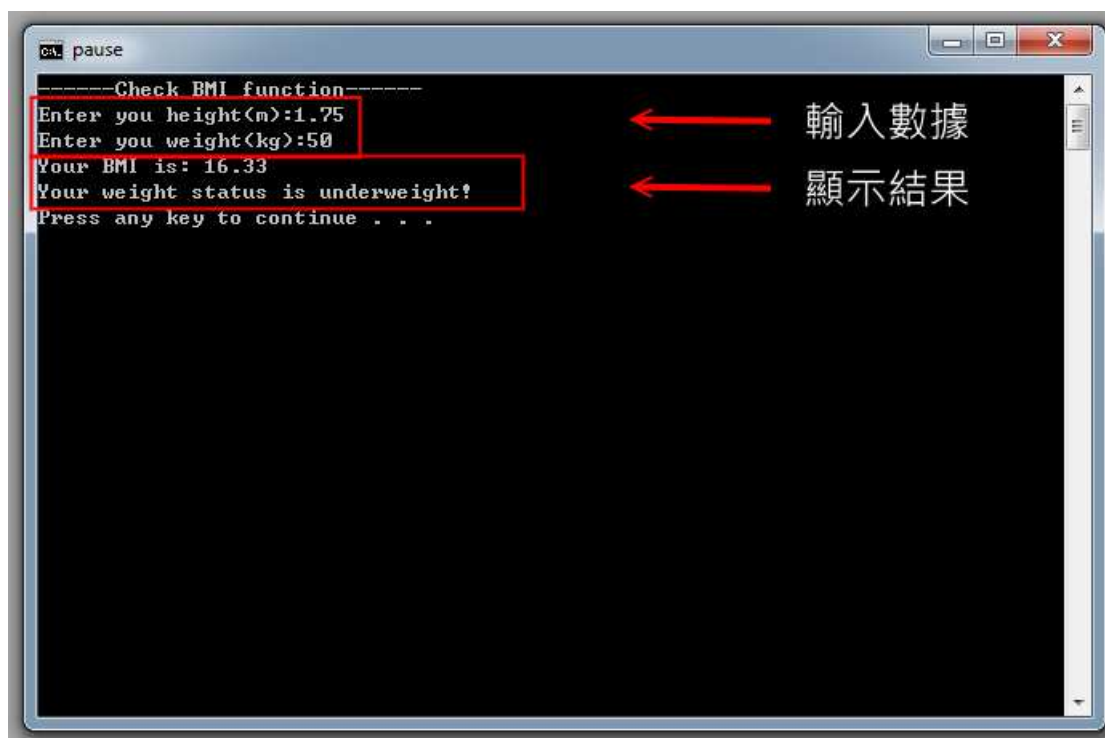
### 2.7.1.6 BMI 計算系統



A screenshot of a Windows command prompt window titled "cmd: pause". The window displays the output of a BMI calculation program. The text is as follows:

```
-----Check BMI function-----  
Enter you height(m):1.75  
Enter you weight(kg):61  
Your BMI is: 19.92  
Your weight status is pass!  
Press any key to continue . . .
```

Two red boxes highlight the input and output sections. A red arrow points from the text "輸入數據" (Input Data) to the first box, which contains the height and weight inputs. Another red arrow points from the text "顯示結果" (Display Result) to the second box, which contains the BMI value and the status message.



A screenshot of a Windows command prompt window titled "cmd: pause". The window displays the output of a BMI calculation program. The text is as follows:

```
-----Check BMI function-----  
Enter you height(m):1.75  
Enter you weight(kg):50  
Your BMI is: 16.33  
Your weight status is underweight!  
Press any key to continue . . .
```

Two red boxes highlight the input and output sections. A red arrow points from the text "輸入數據" (Input Data) to the first box, which contains the height and weight inputs. Another red arrow points from the text "顯示結果" (Display Result) to the second box, which contains the BMI value and the status message.

```
-----Check BMI function-----
Enter you height(m):1.75
Enter you weight(kg):76
Your BMI is: 24.82
Your weight status is overweight!
Press any key to continue . . .
```

← 輸入數據

← 顯示結果

經過測試後，可見系統能準確計算出BMI值及顯示合夥的提示句子。

## 3 評估

在自行進行單元測試後，我便把所有系統功能合併成單一系統，找來三名用戶進行驗收測試。以下將會分別顯示他們的意見。

### 3.1 勵志揚醫生之意見

此系統非常失敗。在「病歷記錄」功能上，除了查看病人的病歷記錄、聯絡方法和家屬資料外，根本就不可以更新資料。與舊系統相比，我還是會選用舊系統。

### 3.2 陳家永先生之意見

系統整體不錯，確定可提供需要的資訊。不過我建議字體要大一點，這樣年紀較老的用戶也能易於閱讀內容。

### 3.3 曾港超先生之意見

系統整體不錯，使用簡單，容易掌握。此外系統的運行速度亦不慢。唯一需要改善的地方是應該加上中文版本，配合不懂英語的人士。

### 3.4 小結

總合以上三位的意見，本系統有三個地方可以改進。

首先是「病歷記錄」功能上，需加上更新資料的功能。第二是字體要大一點，方便年紀較老的用戶閱讀。第三則是要加上中文版本，配合不懂英語的人士。

## **4 系統轉換及維護**

### **4.1 系統轉換**

當醫療資訊系統開發完畢後，並經過嚴格測試，便可以推出使用。系統轉換就是指以新開發的系統取替現有系統的過程。系統轉換的策略可以分為四種：直接切入式轉換、並行式轉換、階段式轉換及引導式轉換，而每種策略都在成本、複雜性和風險等方面也有取捨。

#### **4.1.1 直接切入式轉換**

直接切入式轉換是指新系統會直接被安裝、配置和投入服務。然後原來系統的操作會被終止，新舊兩個系統的操作只有很少、甚至沒有重疊。

直接切入式轉換的優點是其簡單的系統轉換過程，加上新舊兩個系統不會同時進行，所以在軟件、硬件及人力等資源需求也較少。可是這策略的風險十分高，因為萬一新系統失效時，將沒有後備系統作支援的關係。

#### **4.1.2 並行式轉換**

並行式轉換是指新舊兩個系統會在相當一段長時間內同時運作。由於新系統的運作是在現有系統停用前已開始，所以在某時段會出現兩個系統並行運作的情況。亦因如此，在並行運作期間，硬件和人手等資源的使用會倍增（或更高），不過系統失效的風險會被降至最低。

#### **4.1.3 階段式轉換**

階段式轉換是結合直接切入式轉換和並行式轉換兩者的優點，並同時減低整個系統失效的風險。在這策略中，新系統會經過多個階段，一步步取代舊有系統的功能。因此，運作和行政人員有較長時間適應新系統帶來的轉變。可是階段式轉換會增加轉換過程的複雜性，因為它需要把轉換過程劃分為多個階段，產生較多的工作。

#### **4.1.4 引導式轉換**

引導式轉換是指新系統會在整個機構內分部或階段推行。這此階段中，我們會在指定的站內或為指定的用戶群組安裝整個系統。系統會在引導站中進行運作或測試，開發人員也會檢討系統的長處及短處。如果發現系統出現問題，亦能在機構內全面安裝前修正。



與直接切入式轉換相比，引導式轉換的風險較低，因為若新系統失效，也只有引導站受到影響，對整間公司也只會帶來有限度的損失。

	整個組織在同一階段參與		不同用戶於不同階段參與
	新舊系統不會重疊	新舊系統在某段時間同時存在	
一次推出整個系統	直接切入式轉換	並行式轉換	引導式轉換
分階段推出系統部件	階段式轉換		-

最終來說，我建議使用直接切入式轉換，因為在系統發佈之時，而已事先過行多次測試，新系統失效的機會將減到最低。加上用直接切入式轉換，它所需要付出的成本亦比較少。



## **4.2 系統維護**

軟件系統和數據是公司的寶貴資產，所以必須要進行完善的管理及維護工作，防止系統受到破壞和數據被盜等事情發生。

### **4.2.1 登入系統**

本醫療資訊系統居有登入系統。當用戶要使用系統時，便需輸入屬於他們個人的帳戶名稱和密碼。只有他們輸入的資訊正確無誤，才容許他們使用系統，是系統保安的第一道防線。此外，登入系統也是確認用戶身份的一種程序，有雙重的作用。

### **4.2.2 防火牆和防毒軟件**

普遍來說，硬件防火牆有較佳的性能和較高的安全性，主要原因是它能獨立運作，並能隱藏位於硬件防火牆後的伺服器 and 電腦設備，防止他人透過互聯網進行未經授權的存取，甚至可以令伺服器不容易被駭客入侵、破壞或修改伺服器內重要的檔案和數據。

另外，為了防止作為伺服器的電腦受到電腦病毒的感染，所以也需要在伺服器上安裝抗電腦病毒軟件，以防止伺服器因受病毒感染，令儲存在裡面的數據受到毀壞，甚至流失。不過只安裝抗電腦病毒軟件亦不足夠，需要定期更新抗電腦病毒軟件的病毒資料庫，才能減低伺服器感染新型電腦病毒的風險。

### **4.2.3 系統及數據備份**

不管我們做了什麼保安措施，也不能保障不會出現停機的一天。為此，為了減低系統因停機所造成的損失，應定時進行備份工作，比如每數天進行一次。這樣做即使數據流失或系統失效，也能立即復原數據或系統為最近一天內的變更，確保系統能盡快回復原到正常的狀態。若伺服器的負載能力足夠的情況下，便應該每天進行一次備份，進一步減少因停機所帶來的損失。

## 4.2.4 系統的可持續性

由於系統所處的環境可能隨時間而變化，因此用戶的需求也可能有變。因此我們要定期從用戶收集修改要求、監察和改進系統性能、硬件和軟件的升級，以及更新系統文件編製，反映系統的改變。

### 4.2.4.1 收集用戶需求之方法

為收集用戶對系統的修改要求，我建議系統需加上一個公司資訊的功能。比如提供「香城醫療保健網絡」的電話、傳真、電郵地址和郵寄地址等等。此外我們可以加上留言反饋功能，讓用戶直接在系統留下建議，再借互聯網傳送到「香城醫療保健網絡」。這可以讓使用者聯絡我們，為提供改善建議，一些漏洞之處，以及感受。一個良好的回應，可以使我們的香港音樂獎做得更好。

The image displays two screenshots of a website's '联系我们' (Contact Us) page. The top screenshot shows the '留言反馈' (Leave Feedback) form, which includes input fields for '公司:' (Company), '姓名:' (Name), '地址:' (Address), '手机:' (Mobile Phone), and '邮箱:' (Email), followed by a large text area for '留言内容:' (Message Content). At the bottom of the form are '提交' (Submit) and '取消' (Cancel) buttons. The bottom screenshot shows the '联系方式' (Contact Information) section for '深圳市隆信宝商贸有限公司' (Shenzhen Longxinbao Commerce Co., Ltd.), providing contact details: '地址: 中国广东省深圳市罗湖区东门南路3002号华都园大厦12楼D室' (Address: 12/F, D Room, Huadu Garden Building, 3002 Eastmen South Road, Luohu District, Shenzhen, Guangdong Province, China), '邮编: 518001' (Postal Code: 518001), '电话: +86 - 755 - 8226 9177' (Phone: +86 - 755 - 8226 9177), '传真: +86 - 755 - 8226 9105' (Fax: +86 - 755 - 8226 9105), '电邮: info@giantfaith.com' (Email: info@giantfaith.com), '网址: http://www.szgfm.com.cn' (Website: http://www.szgfm.com.cn), and 'MSN: szgfmcl@hotmail.com'.

一些公司的網頁，也會設下留言反饋功能和公司資訊，讓客戶可以與公司聯絡。

## 4.2.4.2 監察系統之方法

在監察系統方法，我建議在系統發生錯誤時，借用互聯網自動傳送使用統計資料及當機報告給「香城醫療保健網絡」。當然，在需要事先得到用戶同意，才能這樣做。

### 隱私權說明

內容設定...

清除瀏覽資料...

Google Chrome 會使用網路服務來改善瀏覽品質。您可以選擇停用這些服務。 [瞭解詳情](#)

- ☒ 使用網路服務來協助解決瀏覽錯誤
- ☒ 使用建議服務，讓系統協助完成您在網址列輸入的搜尋內容或網址
- ☒ 預測網路動作，增進網頁載入效能
- ☒ 阻擋釣魚網站及惡意程式
- ☐ 使用網路服務來協助解決拼字錯誤
- ☐ 自動傳送使用統計資料及當機報告給 Google
- ☐ 將「不追蹤」要求與瀏覽流量一併送出

在 Google Chrome 中，便有用互聯網自動傳送使用統計資料及當機報告的功能。

## 4.3 用戶培訓

為了讓用戶在系統實行時，能發揮更高的生產力，因此需要進行用戶培訓，為用戶提供實際的操作經驗。否則的話，用戶適應新系統的速度會較低，使用時出現錯的機會也會大大更加，令用戶感到困擾，同時新系統將在遠於最佳性能的狀態下運作。

### 4.3.1 用戶培訓的方法

用戶培訓的方法有很多種，比如為不同類別的用戶提供分組培訓，以熟習新系統的功能。此外我們亦需推出系統操作手冊，讓用戶可以自學。另外我們可以使用網上求助台，對用戶提供協助。我們也可以在互聯網建立討論區或討論群組，讓用戶互相分享經驗，以及疑難排解的技巧。

最後，為配合系統維護時對各功能和特點的改動，我們也需不時向用戶提供有關的培訓。

## 5 總結與反思

### 5.1 系統功能比本身的構思少

由於時間有限，要在短短的一年多時間同時完成多份校本評核，因此在編寫系統功能上，編寫出來的功能比本身的構思少。 比如沒有編寫出新增病歷記錄、新增或刪除醫療機構／健康提示資料的功能，甚至後其加上 B M I 計算器時，想出的 B M I 結果統計功能也沒有完成。

### 5.2 對編寫程序語言 Pascal 有更深刻的印象

為了完成這次的校本評核，我先後重新編寫系統最少也有三次，這也是剛才提到時間有限令編寫出來的功能比本身構思的少之原因。 此外，為了編寫系統，我多次翻閱教科書，到互聯網上查閱有關 Pascal 編寫的內容，使我對 Pascal 有更深刻的印象。

## 6 資料來源

醫療通網站

<https://apps.hcv.gov.hk/ZH/Index.htm>

合作計劃-醫療通

[http://www.ehealth.gov.hk/tc/ehr\\_related\\_projects/partnership\\_projects/ehealth\\_system.html](http://www.ehealth.gov.hk/tc/ehr_related_projects/partnership_projects/ehealth_system.html)

醫療保健（大紀元新聞網）

<http://www.epochtimes.com/b5/nsc1002.htm>

編程語言\_百度百科

<http://baike.baidu.com/view/552871.htm>

深圳市隆信寶商貿有限公司

<http://www.szgfm.com.cn/index.aspx>

PEARSON Longman 新高中資訊及通訊科技－軟件開發(選修部分 D1)

PEARSON Longman 新高中資訊及通訊科技－軟件開發(選修部分 D2)

Google Chrome 瀏覽器之 setting 功能

## 附件--醫療資訊系統程式編碼

```
program SBA;
uses crt;
type
  ACPWType=record {帳號及密碼}
    ACCT:string[30]; {Account}
    PSWD:string[30]; {Password}
    CLSS:string[30]; {帳號類別}
  end;
  HTtype=record {健康提示資料}
    topic:string; {主題}
    body:string {內容}
  end;
  MItype=record {醫療機構資料}
    name:string;
    add:string;
    tel:string[8];
    fax:string[8];
    email:string;
    web_add:string
  end;
  CFsNolist=record {病歷記錄號碼資料}
    CFsNo:string;
    name:string;
  end;
  CFsPPType=record {病歷記錄個人資料 Personal Profile}
    CFsNo:string;
    surname:string; {姓}
    givenname:string; {名}
    sex:string[1]; {性別}
    HKID:string[8]; {香港身份証}
    dy:integer; {出生年期}
    dm:integer; {出生月分}
    dd:integer; {出生日期}
  end;
  CFsCIType=record {病歷記錄個人聯絡資料 Communication Information}
    CFsNo:string;
    hp:string[8]; {固網電話}
```

```

        mp:string[8]; {手機號碼}
        add:string;
        add_area:string;
    end;
CFsFIType=record  {病歷記錄家屬資料 Family Information}
    surname:string; {姓}
    givenname:string; {名}
    sex:string[1]; {性別}
    relationship:string;
    hp:string[8]; {固網電話}
    mp:string[8]; {手機號碼}
    add:string;
    add_area:string;
end;
CFsSIType=record  {病歷記錄個人疾病資料 Sick Information}
    SIdate:string;
    SickI:string;
    mdi:string; {medicine}
    slp:string[1]; {sick leave paper}
end;

const
    MITnum=2;
    HTTnum=13;
var
    AC_PW:array of ACPWType;
    HT_file:array[1..HTTnum] of HTtype;
    MI_file:array[1..MITnum] of MItype;
    CFsNL_file:array of CFsNolist;
    CFsPP_file:array of CFsPPType;
    CFsCI_file:array of CFsCIType;
    CFsFI_file:array of CFsFIType;
    CFsSI_file:array of CFsSIType;
    position:integer;
    ch_no:integer;
    chse:integer;
procedure login_function(var index:integer); {登入系統}
var
    ACPWfile:text;
    AC_count:integer;

```

```
ACtarget:string;
PWtarget:string;
found:boolean;
ACcount:integer;
IDnum:integer;
begin
  IDnum:=0;
  AC_count:=0;
  assign(ACPWfile,'ACPW.txt');
  reset(ACPWfile);
  while not eof(ACPWfile) do
    begin
      readln(ACPWfile);
      AC_count:=AC_count+1;
    end;
  AC_count:=AC_count div 3;
  setlength(AC_PW,AC_count);
  close(ACPWfile);
  assign(ACPWfile,'ACPW.txt');
  reset(ACPWfile);
  while not eof(ACPWfile) do
    begin
      IDnum:=IDnum+1;
      with AC_PW[IDnum] do
        begin
          readln(ACPWfile,ACCT);
          readln(ACPWfile,PSWD);
          readln(ACPWfile,CLSS);
        end
      end;
    close(ACPWfile);
    write('Account:');
    readln(ACtarget);
    write('Password:');
    readln(PWtarget);
    found:=false;
    ACcount:=0;
    index:=0;
    repeat
```

```

ACcount:=ACcount+1;
with AC_PW[ACcount] do
  if ACCT = ACtarget
    then begin
      index:=ACcount;
      found:=true
    end
until found or (ACcount = IDnum);
if found=false
  then begin
    writeln('Wrong Account or Password. ');
    writeln('Press any key to continue... ');
    ch_no:=7;
    readln();
  end
else begin
  with AC_PW[index] do
    if PWtarget = PSWD
      then begin
        writeln('Login Successful. ');
        writeln('Press any key to continue... ');
        ch_no:=0;
        readln();
      end
    else begin
      writeln('Wrong Account or Password. ');
      writeln('Press any key to continue... ');
      ch_no:=7;
      readln();
    end;
  end;
end; {end of login_function}
procedure goto_XY; {gotoXY 系統}
var
  i:integer;
begin
  gotoXY(1,1);
  for i:=1 to 50 do
    begin

```



```
writeln(' ');
end;
gotoXY(1,1);
end;
procedure CFsListDataInput; {CFs List Data Input}
var
  NLfile:text;
  NLnum:integer;
  NLcount:integer;
begin
  NLcount:=0;
  NLnum:=0;
  assign(NLFile,'CFsNolist.txt');
  reset(NLFile);
  while not eof(NLFile) do
    begin
      readln(NLFile);
      NLcount:=NLcount+1;
    end;
  NLcount:=NLcount div 2;
  setlength(CFsNL_file,NLcount);
  close(NLfile);
  assign(NLFile,'CFsNolist.txt');
  reset(NLFile);
  while not eof(NLFile) do
    begin
      NLnum:=NLnum+1;
      with CFsNL_file[NLnum] do
        begin
          readln(NLFile,CFsNo);
          readln(NLFile,name);
        end
      end;
    end;
  close(NLfile);
end;
procedure CFsDataInput(case_no:string;var FIcount,SIcount:integer); {CFs Data Input}
var
  PPfile:text;
```

```

Cfile:text;
Ffile:text;
Sfile:text;
Ftemp:string;
Sitemp:string;
PPnum:integer;
CInum:integer;
FInum:integer;
SInum:integer;
PPcount:integer;
CIcon:integer;
begin
  PPcount:=0;
  PPnum:=0;
  assign(PPFile,'CFPP.txt');
  reset(PPFile);
  while not eof(PPFile) do
    begin
      readln(PPFile);
      PPcount:=PPcount+1;
    end;
  PPcount:=PPcount div 8;
  setlength(CFsPP_file,PPcount);
  close(PPfile);
  assign(PPFile,'CFPP.txt');
  reset(PPFile);
  while not eof(PPFile) do
    begin
      PPnum:=PPnum+1;
      with CFsPP_file[PPnum] do
        begin
          readln(PPFile,CFsNo);
          readln(PPFile,surname);
          readln(PPFile,givenname);
          readln(PPFile,sex);
          readln(PPFile,HKID);
          readln(PPFile,dy);
          readln(PPFile,dm);
          readln(PPFile,dd);

```

```
end
end;
close(PPfile);
CIcount:=0;
CInum:=0;
assign(CIfile,'CFCL.txt');
reset(CIFile);
while not eof(CIFile) do
begin
  readln(CIFile);
  CIcount:=CIcount+1;
end;
CIcount:=CIcount div 5;
setlength(CFsCI_file,CIcount);
close(CIfile);
assign(CIFile,'CFCL.txt');
reset(CIFile);
while not eof(CIFile) do
begin
  CInum:=CInum+1;
  with CFsCI_file[CInum] do
  begin
    readln(CIFile,CFsNo);
    readln(CIFile,hp);
    readln(CIFile,mp);
    readln(CIFile,add);
    readln(CIFile,add_area);
  end
end;
close(CIFile);
FIcount:=0;
FInum:=0;
FItemp:=case_no+'CFsFI.txt';
assign(FIFile,FItemp);
reset(FIFile);
while not eof(FIFile) do
begin
  readln(FIFile);
  FIcount:=FIcount+1;
```

```

end;
FIcount:=FIcount div 8;
setlength(CFsFI_file,FIcount);
close(FIfile);
assign(FIFile,FItemp);
reset(FIFile);
while not eof(FIFile) do
begin
  FInum:=FInum+1;
  with CFsFI_file[FInum] do
  begin
    readln(FIFile,surname);
    readln(FIFile,givenname);
    readln(FIFile,sex);
    readln(FIFile,relationship);
    readln(FIFile,hp);
    readln(FIFile,mp);
    readln(FIFile,add);
    readln(FIFile,add_area);
  end
end;
close(FIFile);
SIcount:=0;
SInum:=0;
SItemp:=case_no+'CFsSI.txt';
assign(SIFile,SItemp);
reset(SIFile);
while not eof(SIFile) do
begin
  readln(SIFile);
  SIcount:=SIcount+1;
end;
SIcount:=SIcount div 4;
setlength(CFsSI_file,SIcount);
close(SIfile);
assign(SIFile,SItemp);
reset(SIFile);
while not eof(SIFile) do
begin

```

```
SInum:=SInum+1;
with CFsSI_file[SInum] do
begin
  readln(SIFile,sIdate);
  readln(SIFile,SickI);
  readln(SIFile,mdi);
  readln(SIFile,slp);
end
end;
close(SIFile);
end;
procedure CFsPP(num:integer); {CFs Personal Profile}
begin
  goto_XY;
  writeln('-----Personal Profile-----');
  with CFsPP_file[num] do
  begin
    writeln('Case File no. : ',CFsNo);
    writeln('Name : ',surname, ' ',givenname);
    writeln('Sex: ',sex);
    writeln('HKID : ',HKID);
    writeln('Day of birthday : ',dd, '/', dm, '/', dy);
    writeln();
    writeln('Press any key to continue...');
    readln();
    goto_XY;
  end;
end;
procedure CFsCI(num:integer); {CFs Communication Information}
begin
  goto_XY;
  writeln('-----PCommunication Information-----');
  with CFsPP_file[num] do
  begin
    writeln('Case File no. : ',CFsNo);
    writeln('Name : ',surname, ' ',givenname);
  end;
  with CFsCI_file[num] do
  begin
```

```

        writeln('Home Phone No. : ',hp);
        writeln('Personal Mobile Phone No. : ',mp);
        writeln('Address : ',add, ' ',add_area);
        writeln();
        writeln('Press any key to continue...');
        readln();
        goto_XY;
    end;
end;
procedure CFsFI(num,total_no:integer); {CFs Family Information}
var
    FIfile:text;
    i:integer;
begin
    goto_XY;
    writeln('-----Family Information-----');
    with CFsPP_file[num] do
        begin
            writeln('Case File no. : ',CFsNo);
            writeln('Name : ',surname, ' ',givenname);
            writeln();
        end;
        num:=0;
        for i:=1 to total_no do
            begin
                num:=num+1;
                with CFsFI_file[num] do
                    begin
                        writeln('Name : ',surname, ' ',givenname);
                        writeln('Sex : ',sex);
                        writeln('Relationship:',relationship);
                        writeln('Home Phone No. : ',hp);
                        writeln('Personal Mobile Phone No. : ',mp);
                        writeln('Address : ',add, ' ',add_area);
                        writeln();
                    end;
                end;
            end;
        writeln('Press any key to continue...');
        readln();
    end;
end;

```

```
goto_XY;
end;
procedure CFsSI(num,total_no:integer); {CFs Sick Information}
var
  FIfile:text;
  i:integer;
  CFs_No:string;
  nothinguse:integer;
begin
  goto_XY;
  writeln('-----Family Information-----');
  with CFsPP_file[num] do
    begin
      writeln('Case File no. : ',CFsNo);
      writeln('Name : ',surname,' ',givenname);
      writeln();
    end;
    str(num,CFs_No);
    CFsDataInput(CFs_No,total_no,nothinguse);
    num:=0;
    for i:=1 to total_no do
      begin
        num:=num+1;
        with CFsFI_file[num] do
          begin
            writeln('Name : ',surname,' ',givenname);
            writeln('Sex : ',sex);
            writeln('Relationship: ',relationship);
            writeln('Home Phone No. : ',hp);
            writeln('Personal Mobile Phone No. : ',mp);
            writeln('Address : ',add,' ',add_area);
            writeln();
          end;
        end;
      end;
      writeln('Press any key to continue...');
      readln();
      goto_XY;
    end;
  procedure CFsChse; {CFs Choose List}
```

```

var
CF_no:integer;
ch_no:integer;
C_no:string;
FI_num:integer;
SI_num:integer;
begin
    goto_XY;
    CF_no:=0;
    CFsListDataInput;
    writeln('-----Look at the case file-----');
    write('Please enter the Case File no. :');
    readln(CF_no);
    goto_XY;
    str(CF_no,C_no);
    CFsDataInput(C_no,FI_num,SI_num);
    repeat
        with CFsNL_file[CF_no] do
            begin
                writeln('Case File no. :',CFsNo);
                writeln('Name :',name);
                writeln(' ');
                writeln('1.Personal Profile');
                writeln('2.Communication Information');
                writeln('3.Family Information');
                writeln('4.Sick Information');
                writeln('5.Exit. ');
                write('Please choose the function: ');
                readln(ch_no);
                goto_XY;
                case ch_no of
                    1:CFsPP(CF_no);
                    2:CFsCI(CF_no);
                    3:CFsFI(CF_no,FI_num);
                    4:CFsSI(CF_no,SI_num);
                    5:writeln('OK. ');
                else writeln('ERROR!Please enter again. ');
                end;
            end;
        end;
    end;
end;

```



```
until ch_no=5;
end;
procedure CFs; {CFs Main Program}
var
  CFsnum:real;
  ch_no:integer;
begin
  repeat
    goto_XY;
    writeln('-----Case Files-----');
    writeln('1.Look at the case file. ');
    writeln('2.Exit');
    write('Please choose the function: ');
    readln(ch_no);
    case ch_no of
      1:CFsChse;
      2:writeln('Okay. ');
    else begin
      writeln('ERROR!Please enter again. ');
      writeln('Press any key to continue... ');
      readln();
    end;
  end;
  until ch_no=2;
end;
procedure checkBMI; {BMI 計算系統}
var
  height,weight,bmi:real;
begin
  goto_XY;
  writeln('-----Check BMI function-----');
  write('Enter you height(m): ');
  readln(height);
  write('Enter you weight(kg): ');
  readln(weight);
  bmi:= weight/(sqr(height));
  writeln('Your BMI is: ',bmi:1:2);
  if BMI<18.5 then
    writeln('Your weight status is underweight!')
```

```

else
  if (BMI>24.00) and (BMI<28.00) then
    writeln('Your weight status is overweight!')
  else
    if BMI>28.00 then
      writeln('Your weight status is obese!')
    else
      writeln('Your weight status is pass!');
writeln(' ');
writeln('Press any key to continue...');
readln();
goto_XY;
end;
procedure HTread(ch_no:integer); {健康提示系統}
var
  DiskFile:text;
  ch_num:integer;
  exit_no:integer;
  count:integer;
begin
  goto_XY;
  case ch_no of
    1:assign(DiskFile,'Hair_loss.txt');
    2:assign(DiskFile,'Male_diet.txt');
    3:assign(DiskFile,'Male_sports.txt');
    4:assign(DiskFile,'Mens_Health.txt');
    5:assign(DiskFile,'Womens_Health.txt');
  end;
  reset(DiskFile);
  count:=0;
  while not eof(DiskFile) do
    begin
      count:=count+1;
      with HT_file[count] do
        begin
          readln(DiskFile,topic);
          readln(DiskFile,body);
        end;
      end;
    end;
end;

```

```
repeat
  count:=0;
  repeat
    count:=count+1;
    with HT_file[count] do
      begin
        writeln(count,'.',topic);
      end;
  until count=HTTnum;
  exit_no:=count+1;
  writeln(exit_no,'.Exit. ');
  ch_num:=0;
  write('Your choose is: ');
  readln(ch_num);
  goto_XY;
  if ch_num<>exit_no then
    begin
      with HT_file[ch_num] do
        begin
          writeln('Topic: ',topic);
          writeln('Body: ',body);
          writeln(' ');
          writeln('Press any key to continue...');
          readln();
          goto_XY;
        end;
      end;
    until ch_num=exit_no;
    goto_XY;
  end; {HTread}
procedure HTinput(ch_no:integer); {健康提示系統}
begin
  goto_XY;
  HTread(ch_no);
  goto_XY;
end;
procedure HTs; {健康提示選擇目錄系統}
var
  HT_ch:integer;
```

```

begin
  goto_XY;
  repeat
    HT_ch:=0;
    writeln('-----Health Tips-----');
    writeln('1.Hair Loss');
    writeln('2.Male Diet');
    writeln('3.Male Sports');
    writeln('4.Men"s Health');
    writeln('5.Women"s Health');
    writeln('6.Check BMI');
    writeln('7.Exit. ');
    write('Your choose is: ');
    readln(HT_ch);
    goto_XY;
    case HT_ch of
      1:HTinput(HT_ch);
      2:HTinput(HT_ch);
      3:HTinput(HT_ch);
      4:HTinput(HT_ch);
      5:HTinput(HT_ch);
      6:checkBMI;
    end;
  until HT_ch=7;
  goto_XY;
end;
procedure Mlread(ch_no:integer); {醫療機構系統}
var
  DiskFile:text;
  ch_num:integer;
  exit_no:integer;
  count:integer;
begin
  goto_XY;
  case ch_no of
    1:assign(DiskFile,'PHI.txt');
    2:assign(DiskFile,'PH.txt');
  end;
  reset(DiskFile);

```

```
count:=0;
while not eof(DiskFile) do
begin
    count:=count+1;
    with MI_file[count] do
    begin
        readln(DiskFile,name);
        readln(DiskFile,add);
        readln(DiskFile,tel);
        readln(DiskFile,fax);
        readln(DiskFile,email);
        readln(DiskFile,web_add);
    end;
end;
repeat
    count:=0;
    repeat
        count:=count+1;
        with MI_file[count] do
        begin
            writeln(count,'.',name);
        end;
    until count=MITnum;
    exit_no:=count+1;
    writeln(exit_no,'.Exit. ');
    ch_num:=0;
    write('Your choose is: ');
    readln(ch_num);
    if ch_num<>exit_no then
    begin
        with MI_file[ch_num] do
        begin
            goto_XY;
            writeln('Name: ',name);
            writeln('Add.: ',add);
            writeln('Tel.: ',tel);
            writeln('Fax.: ',fax);
            writeln('E-Mail: ',email);
            writeln('Web Add.: ',web_add);
```

```

        writeln(' ');
        writeln('Press any key to continue...');
        readln();
    end;
    end;
    goto_XY;
until ch_num=exit_no;
goto_XY;
end; {MIread}
procedure MInput(ch_no:integer); {醫療機構系統}
begin
    goto_XY;
    MIread(ch_no);
    goto_XY;
end;
procedure MIs; {醫療機構系統}
var
    MI_ch:integer;
begin
    goto_XY;
    repeat
        MI_ch:=0;
        writeln('-----Medical Institutions-----');
        writeln('1.Public Hospitals And Institutions');
        writeln('2.Private Hospitals');
        writeln('3.Exit. ');
        write('Your choose is: ');
        readln(MI_ch);
        writeln(' ');
        case MI_ch of
            1:MIinput(MI_ch);
            2:MIinput(MI_ch);
            else if (MI_ch<1) or (MI_ch>3)
                then writeln('Error!');
        end;
        until MI_ch=3;
        goto_XY;
    end;
end;
procedure ch_funtn(index:integer); {主目錄系統}

```

```
begin
  goto_XY;
  ch_no:=0;
  with AC_PW[index] do
    if CLSS = 'doctor'
      then begin
        writeln('-----Main List-----');
        writeln('1.Case Files. ');
        writeln('2.Medical Institutions. ');
        writeln('3.Health Tips. ');
        writeln('4.Exit. ');
        write('Please choose the function: ');
        readln(ch_no);
        case ch_no of
          1:CFs;
          2:MIIs;
          3:HTs;
          4:ch_no:=10;
          else writeln('error!');
        end;
        goto_XY;
      end
    else begin
        writeln('-----Main List-----');
        writeln('1.Medical Institutions. ');
        writeln('2.Health Tips. ');
        writeln('3.Exit. ');
        write('Please choose the function: ');
        readln(ch_no);
        case ch_no of
          1:MIIs;
          2:HTs;
          3:ch_no:=10;
          else writeln('error!');
        end;
        goto_XY;
      end;
    end;
  end;
begin {main program}
```

```
ch_no:=0;  
repeat  
  login_function(position);  
  goto_XY;  
until ch_no=0;  
  repeat  
    goto_XY;  
    ch_funtn(position);  
  until ch_no=10;  
end.
```