

人工智慧與軟體工程之應用

國立彰化師範大學資訊工程系

賴聯福



Outline

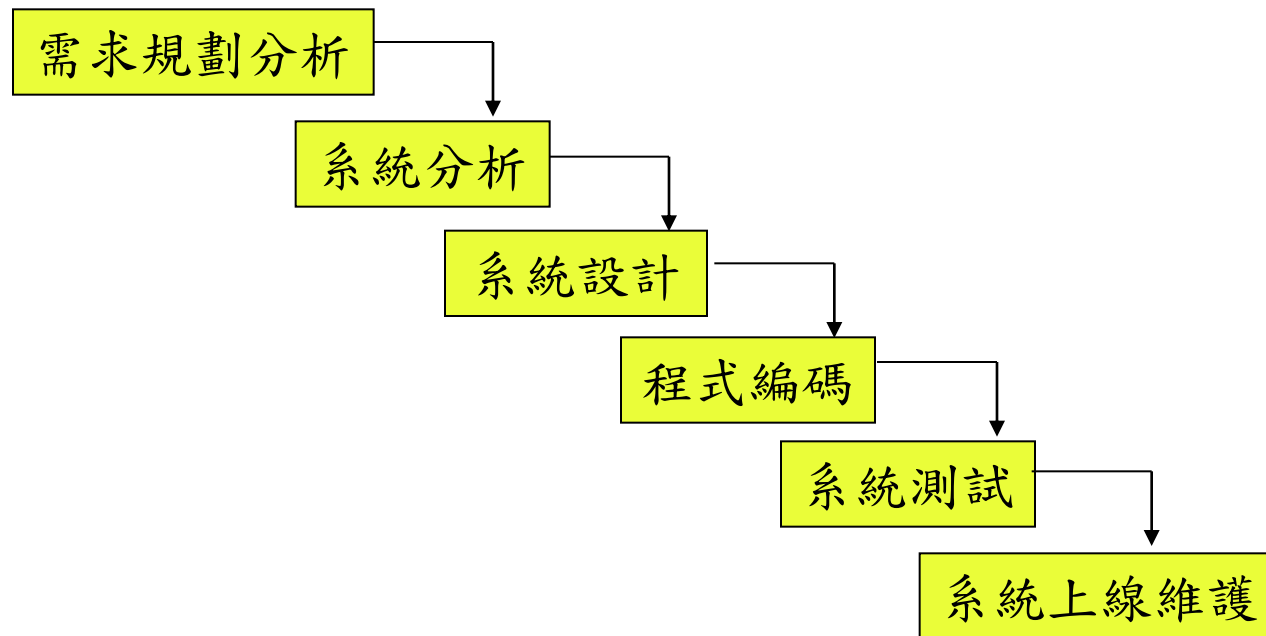
- 軟體工程 (Software Engineering)
- 人工智慧 (Artificial Intelligence)
- 人工智慧的應用
- 人工智慧的執行
- 應用人工智慧技術於軟體工程
- 結論



軟體工程 (Software Engineering)

- 軟體工程

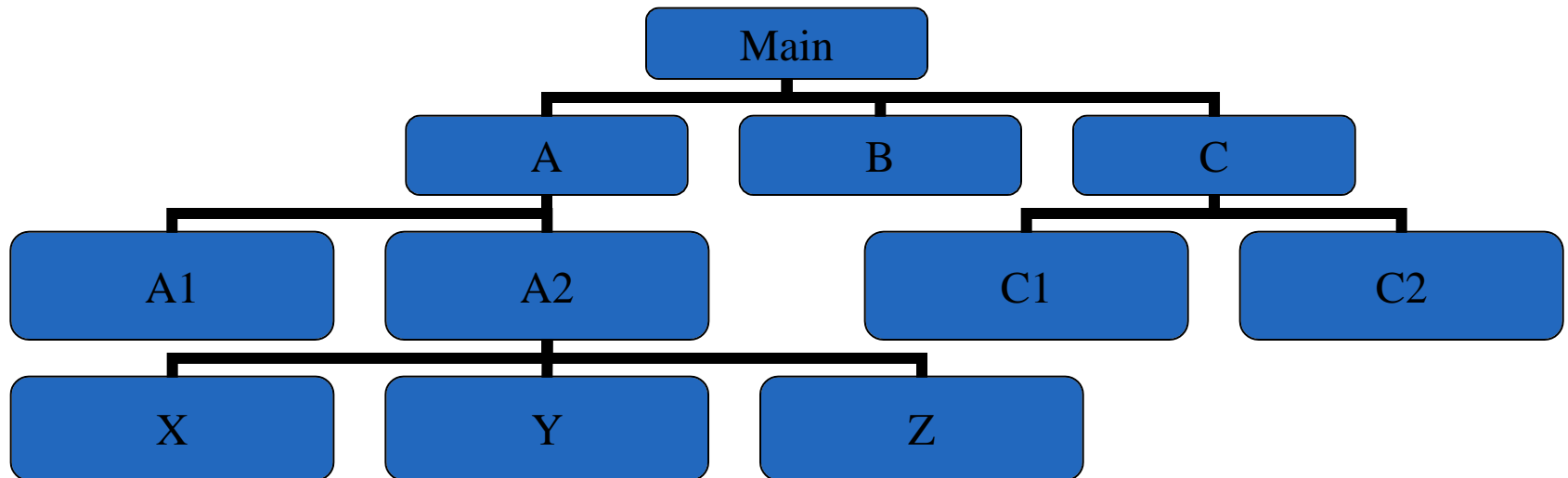
- 以標準的工程方法分多個階段一步步來開發軟體



結構化系統分析

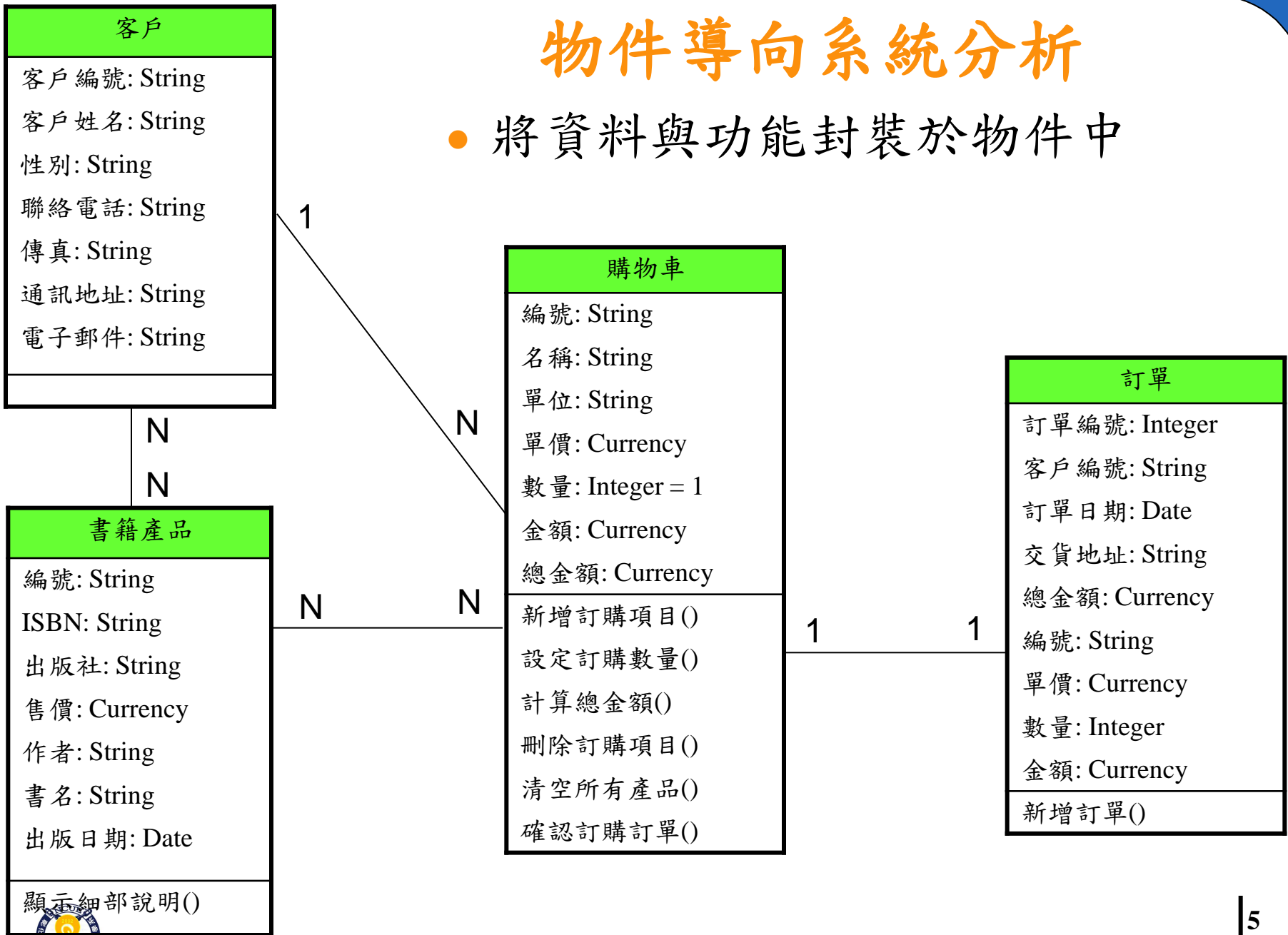
- 問題分解 / 功能分解

- 將大而複雜的問題(或功能)分解到較小而簡化的問題(或功能)，直到成為容易理解的問題(或可直接實作的功能)為止



物件導向系統分析

- 將資料與功能封裝於物件中



人工智慧 (Artificial Intelligence)

- 電腦聰明？有智慧？
 - 電腦會下棋、算命、解數學難題、玩電腦遊戲、看病、問題診斷等
 - 智慧功能由軟體程式所提供
 - 軟體程式所提供的功能，電腦才會具有此能力，電腦並不會自己發展新功能
- 一般常用來撰寫人工智慧軟體之語言
 - 人工智慧程式語言
 - Prolog、Lisp、CLIPS 等
 - 一般用途程式語言
 - VB、C、JAVA 等



人工智慧 (AI) 的意義

- Using methods based on the intelligent behavior of humans and other animals to solve complex problems

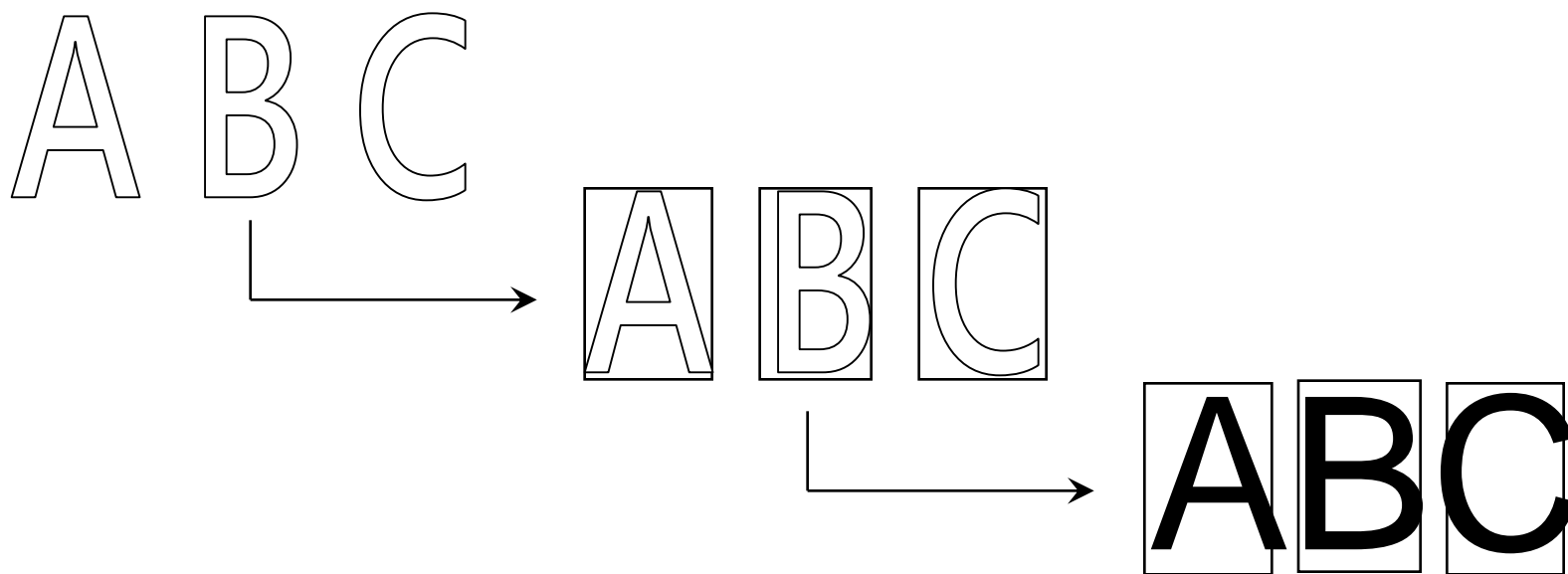
-- Ben Coppin

- 人類的智慧行為有哪些？
 - 理解能力
 - 圖形識別（視覺）、聲音識別（聽覺）、語言理解
 - 學習能力
 - 從經驗中學習、從範例中學習、從教導中學習
 - 推理能力
 - 邏輯推理、知識推理
 - 判斷能力
 - 診斷、解釋、預測
 - 求解能力
 - 下棋、問題規劃



人工智慧的應用：圖形識別（電腦視覺）

- 以文字辨別為例，需三個步驟
 - 方格切割
 - 骨架化
 - 資料庫比對



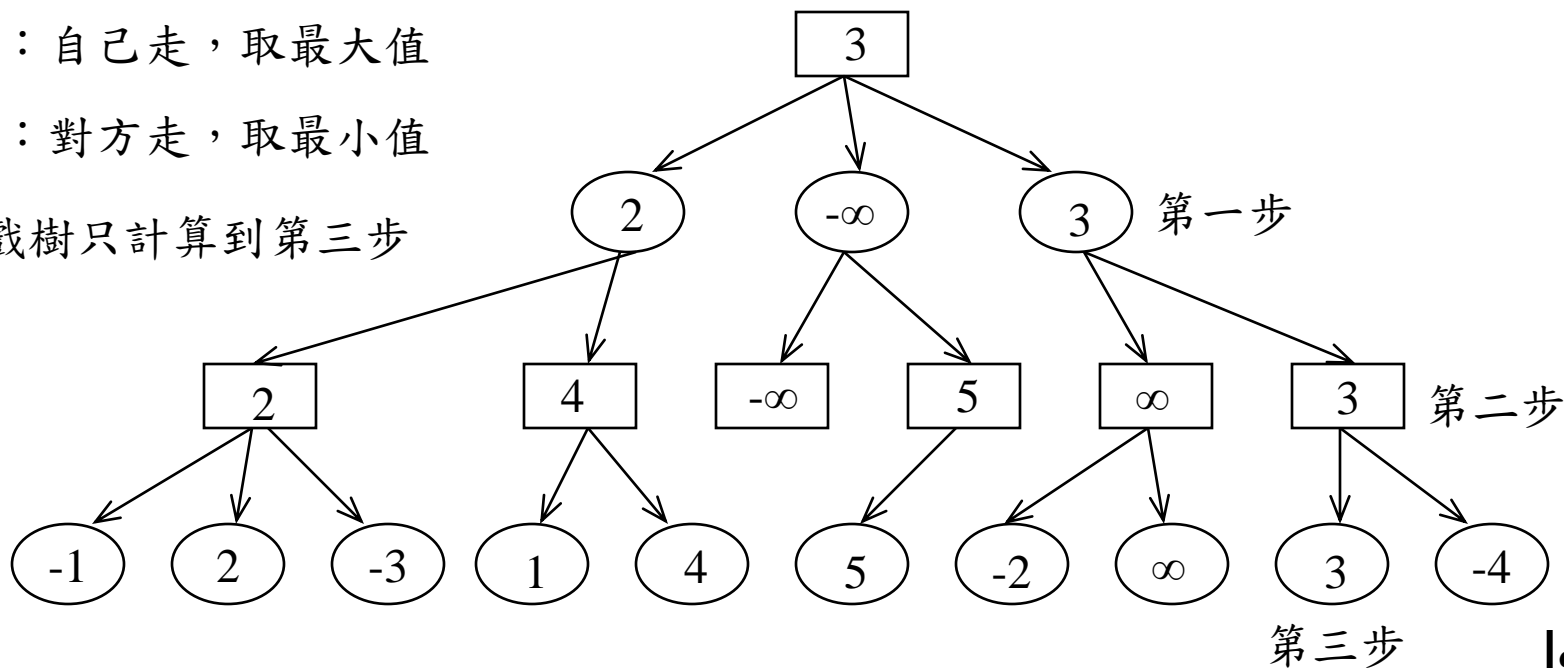
人工智慧的應用：下棋

- 兩人輪流下棋，可以用遊戲樹（game tree）來計算對自己最有利的走法
 - 訂出評估函數：計算各種走法的評估值，以選擇對自己最有利的走法
 - MinMax search：由於為兩人輪流下棋，自己會選擇下層評估值的最大值（對自己最有利），而對方會選擇最小值（對自己最不利）

□：自己走，取最大值

○：對方走，取最小值

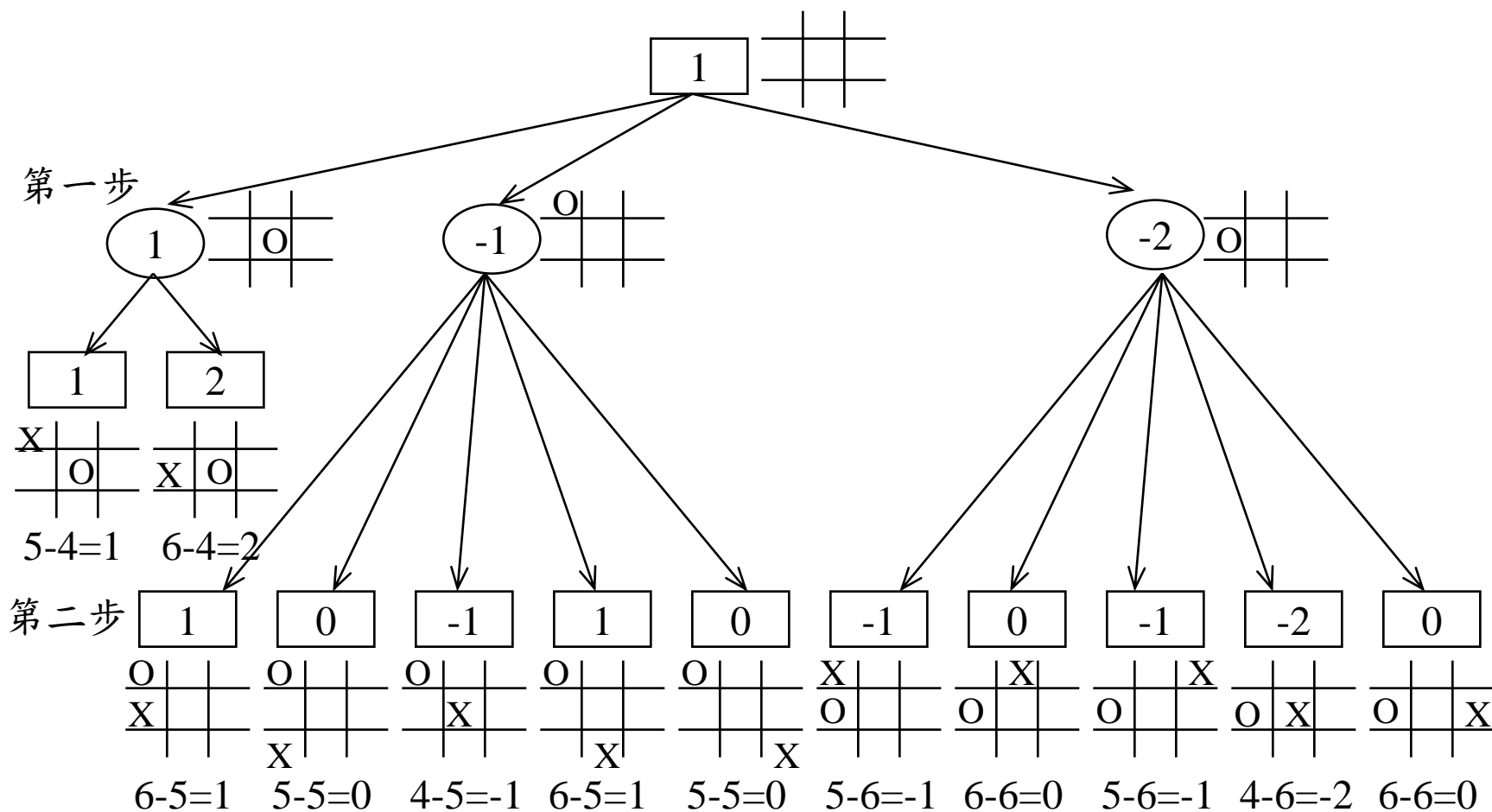
此遊戲樹只計算到第三步



● 以圈叉棋為例：假設只計算兩步

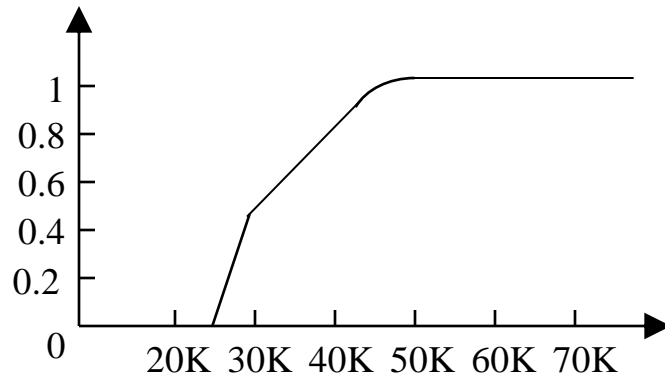
– 自己贏（連成一線）則評估值= ∞ ，對方贏則評估值= $-\infty$

– 否則 評估值=自己還有可能連線的總數－對方還有可能連線的總數

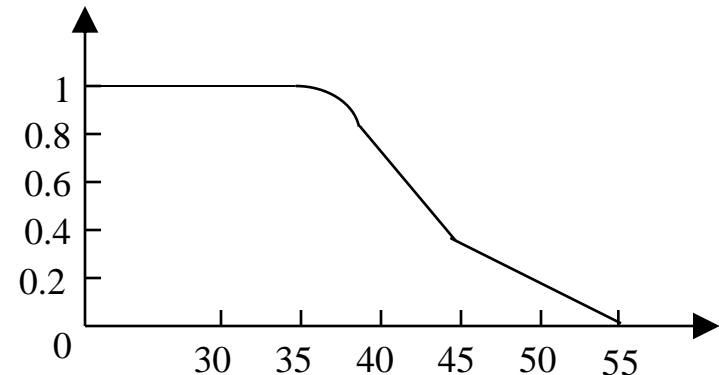


人工智慧的應用：模糊邏輯 (Fuzzy Logic)

- 一般日常用語或需求經常都是模糊的而非明確絕對的，例如
 - 冷氣再冷一點
 - 電子鍋煮的飯稍微硬一點
 - 身材高一點、年紀輕一點、快、慢、多、少、輕、重、冷、熱等等
- 傳統邏輯的真值只有“真”（1）或“假”（0），模糊邏輯允許真值介於0與1之間，而非絕對
 - 模糊邏輯可以使用成員函數（membership function）來表達模糊的意義
 - 例如，某人找工作的條件為錢多、事少、離家近，其成員函數



薪水多一點的成員函數



每週工作時數少一點的成員函數



人工智慧的應用：專家系統 (Expert Systems)

- 利用 專家知識 來診斷、解釋、預測、或求解答
 - 以醫生診斷系統為例：
 - 將醫生的專家知識電腦化，儲存在知識庫 (Knowledge Base) 中
 - 若出現頭痛與鼻塞症狀，則為 A 型感冒症狀
 - 若出現咳嗽、流鼻水、與打噴嚏症狀，則為 B 型感冒症狀
 - 若出現發燒與喉嚨痛症狀，則為 C 型感冒症狀
 - 若為 A 型感冒症狀需吃阿司匹靈
 - 若有胃痛症狀需吃胃藥
 - 病人可以輸入其身體狀況到此醫生診斷系統中
 - 假設輸入頭痛、鼻塞、與胃痛三種症狀
 - 專家系統可以根據所輸入的事實來比對 (match) 專家知識，以自動推理出診斷結果與藥方
 - 診斷結果：A 型感冒症狀
 - 藥方：阿司匹靈與胃藥



人工智慧的執行

- 使用人工智慧方法求得問題解答的過程中，需要執行兩個重要工作
 - 知識的表示
 - 知識的推理
- 數學邏輯為最常用的知識表示法
 - 將知識表示成數學邏輯後，可利用數學推演的證明，從舊有知識中推理出新知識



述語邏輯 (Predicate Logic) 之例子

– 假設我們有下列五項知識與事實，請問 Mary 與 Peter 的關係

- 若 x 是 y 的父親， y 是 z 的父親，則 x 是 z 的祖父
- 若 x 是 y 的祖父， z 是 x 的太太，則 z 是 y 的祖母
- John 是 Kevin 的父親
- Kevin 是 Peter 的父親
- Mary 是 John 的太太

– 把這五項知識與事實表示成述語邏輯

- $(\forall x)(\forall y)(\forall z) \text{Father}(x,y) \wedge \text{Father}(y,z) \rightarrow \text{GrandFather}(x,z)$
- $(\forall x)(\forall y)(\forall z) \text{GrandFather}(x,y) \wedge \text{Wife}(z,x) \rightarrow \text{GrandMother}(z,y)$
- $\text{Father}(\text{John}, \text{Kevin})$
- $\text{Father}(\text{Kevin}, \text{Peter})$
- $\text{Wife}(\text{Mary}, \text{John})$



– 利用述語邏輯的歸結推演，以比對匹配及取代的方式產生新知識

- GrandFather(John,Peter)

- $(\forall x)(\forall y)(\forall z) \text{Father}(x,y) \wedge \text{Father}(y,z) \rightarrow \text{GrandFather}(x,z)$

- $\text{Father}(\text{John},\text{Kevin})$

- $\text{Father}(\text{Kevin},\text{Peter})$

- John 取代 x ，Kevin 取代 y ，Peter 取代 z

- GrandMother(Mary,Peter)

- $(\forall x)(\forall y)(\forall z) \text{GrandFather}(x,y) \wedge \text{Wife}(z,x) \rightarrow \text{GrandMother}(z,y)$

- $\text{Wife}(\text{Mary},\text{John})$

- $\text{GrandFather}(\text{John},\text{Peter})$ 為新產生知識

- John 取代 x ，Peter 取代 y ，Mary 取代 z

– 求得 Mary 與 Peter 的關係為 $\text{GrandMother}(\text{Mary},\text{Peter})$

- $\text{GrandMother}(\text{Mary},\text{Peter})$ 稱為一個述語 predicate

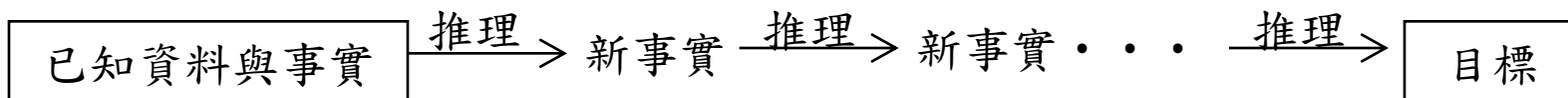
- GrandMother 為後面兩個參數的關係名稱



知識推理方式

- 正向推理 (forward chaining)

- 根據已知的資料與事實 往前推論，產生新的知識，以達到目標
- 或稱為 資料驅動 (data-driven)



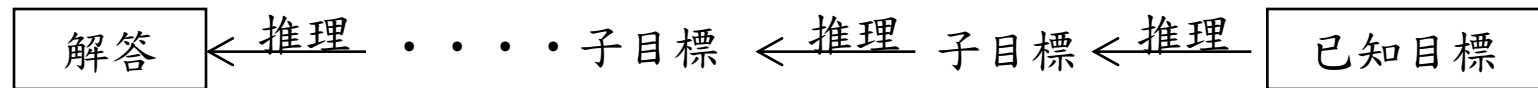
- 述語邏輯的例子即為正向推理

- 已知 3 項事實，包括 John 是 Kevin 的父親、Kevin 是 Peter 的父親、Mary 是 John 的太太、以及祖父關係與祖母關係的兩個規則
- 目標為 “求出 Mary 與 Peter 的關係” (此目標為未知)
- 經由正向推理之後產生新知識，得知 Mary 為 Peter 的祖母



知識推理方式

- 反向推理（backward chaining）
 - 已知目標是什麼，往後推論此目標的達成方法
 - 或稱為 目標驅動（goal-driven）



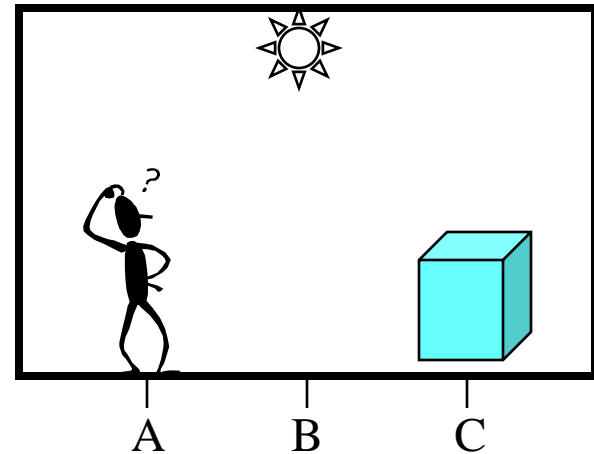
● 反向推理之例子：機器人取下燈泡問題

— 如下圖，房間內有一個機器人、一個箱子、和一個燈泡，燈泡掛在天花板下，但機器人的高度構不到燈泡，請問機器人如何取下燈泡？

- 已知目標為 “機器人取下燈泡”

- 機器人可以做的動作包括

- 機器人用機器手臂取下燈泡
- 機器人從 A 走到 B
- 機器人從 B 走到 C
- 機器人從 C 走到 B
- 機器人從 B 走到 A
- 機器人爬到箱子上
- 機器人把箱子從 A 推到 B
- 機器人把箱子從 B 推到 C
- 機器人把箱子從 C 推到 B
- 機器人把箱子從 B 推到 A



- 請問機器人要如何做（做哪些動作以及順序為何）才能取下燈泡？



● 反向推理過程

- 已知目標為 “機器人取下燈泡” ，可以分解為三個子目標
 - 箱子在 B
 - 機器人站在箱子上
 - 機器人用機器手臂取下燈泡
- “箱子在 B” 的子目標可以再分解為二個子目標
 - 機器人在 C
 - 機器人把箱子從 C 推到 B
- “機器人在 C” 的子目標可以再分解為二個子目標
 - 機器人從 A 走到 B
 - 機器人從 B 走到 C
- 因此解答為五個步驟：
 - 機器人從 A 走到 B、機器人從 B 走到 C、機器人把箱子從 C 推到 B、機器人爬到箱子上、機器人用機器手臂取下燈泡



應用人工智慧技術於軟體工程

● Knowledge-Based Software Engineering

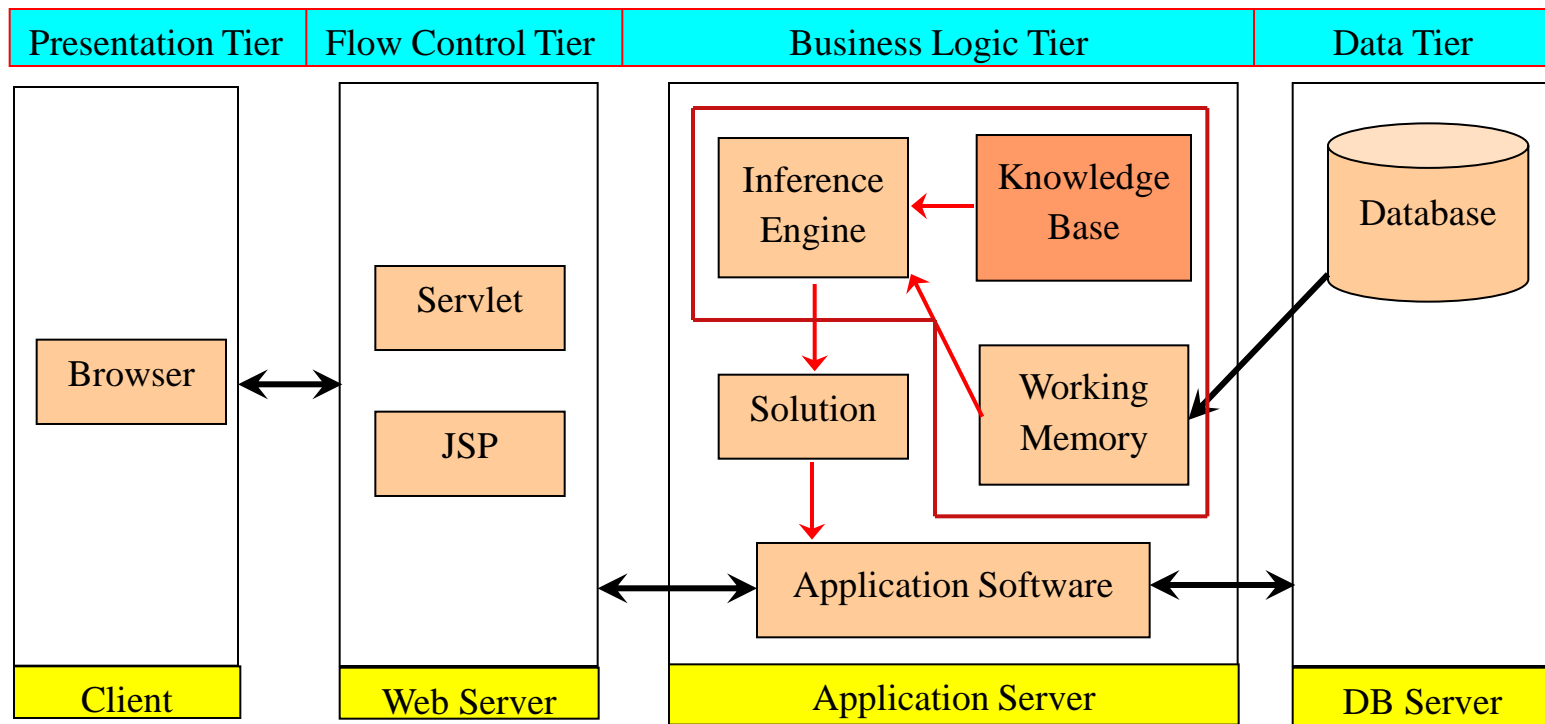
- 提供軟體系統**知識推理**之能力
- 傳統的軟體工程無法處理知識密集(knowledge-intensive)之應用，也無法提供推理功能(inference)以解決問題

● 例子: 資工系排課系統

- 將系辦助理的排課知識寫成專家系統的規則(專家知識庫)
- 載入所有的課程、教師、班級、教室、時段資料(事實)
- 專家系統的推理引擎可以根據所輸入的事實來比對(match)專家知識，以自動推理出同時符合的可行解



將人工智慧程式整合至排課系統



教師開課子系統

國立彰化師範大學
資訊工程學系

排課系統

填寫個人時段

開課

新增選修科目

查詢開課成功

個人功課表

更改密碼

使用者手冊

登出

填寫個人時段

嚴茂旭老師您好：

- 請在下方表格勾選您每個禮拜的喜愛和拒絕的時段，本系統將盡一切可能遵守您的時段安排！
喜愛時段可以無上限勾選，拒絕時段不得超出 28 堂！

步驟一：選擇輸入

步驟二：勾選時段(已勾選之拒絕時段數:15),選擇星期或節之欄位有全選功能

節\日	星期一	星期二	星期三	星期四	星期五
1	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
5	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
6	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
7	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
8	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
9	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
10	X <input type="checkbox"/>	X <input type="checkbox"/>	X <input type="checkbox"/>	X <input type="checkbox"/>	X <input type="checkbox"/>
11	X <input type="checkbox"/>	X <input type="checkbox"/>	X <input type="checkbox"/>	X <input type="checkbox"/>	X <input type="checkbox"/>
12	X <input type="checkbox"/>	X <input type="checkbox"/>	X <input type="checkbox"/>	X <input type="checkbox"/>	X <input type="checkbox"/>

- 並選擇個人喜愛之安排型態：

目前選擇狀態：尚未設定

選擇排課集中代表您想將所有課程集中排在每週一或兩天；而排課分散即代表課程盡量分散成每天都有課，上課時間不超過 4 小時



系辦排課子系統

國立彰化師範大學
資訊工程學系

排課系統

排課維護

- 科目時段存/取
- 自動排課
- 匯入排課結果
- 課表查詢
- 失敗排課列表
- 教師授課時數分配表
- 開設科目表

資工二

step 1 資料維護

- * 科目維護
- * 教師維護
- * 班級維護
- * 教室維護
- * 教室型態維護

step 2 準備開課

- * 開課維護
- * 群組維護
- * 群組開課維護
- * 科目時段維護
- * 教師開課維護

step 3 進行排課

- * 開課維護
- * 排課維護

系統維護

- * 帳戶維護
- * 重新設定
- * 使用者手冊
- * FAQ
- * 登出

列印

資工二課表

節/日	時間	星期一	星期二	星期三	星期四	星期五	星期六
1	08:10 09:00	組合語言與系統程式					
2	09:05 09:55	組合語言與系統程式			計算機組織		
3	10:15 11:05	組合語言與系統程式	體育(四)		計算機組織	通識(四)	
4	11:10 12:00		體育(四)		計算機組織	通識(四)	
5	13:00 13:50	離散數學	硬體描述語言	計算機網路 32310	資料庫系統	微算機技術 32310	
6	13:55 14:45	離散數學	硬體描述語言	計算機網路 32310	資料庫系統	微算機技術 32310	



應用人工智慧技術於軟體工程

- Automated Software Engineering

- 模擬軟體工程師與程式設計師的專家知識

- 軟體開發自動化

- 軟體規格自動驗證

- 軟體規格→自動產生程式

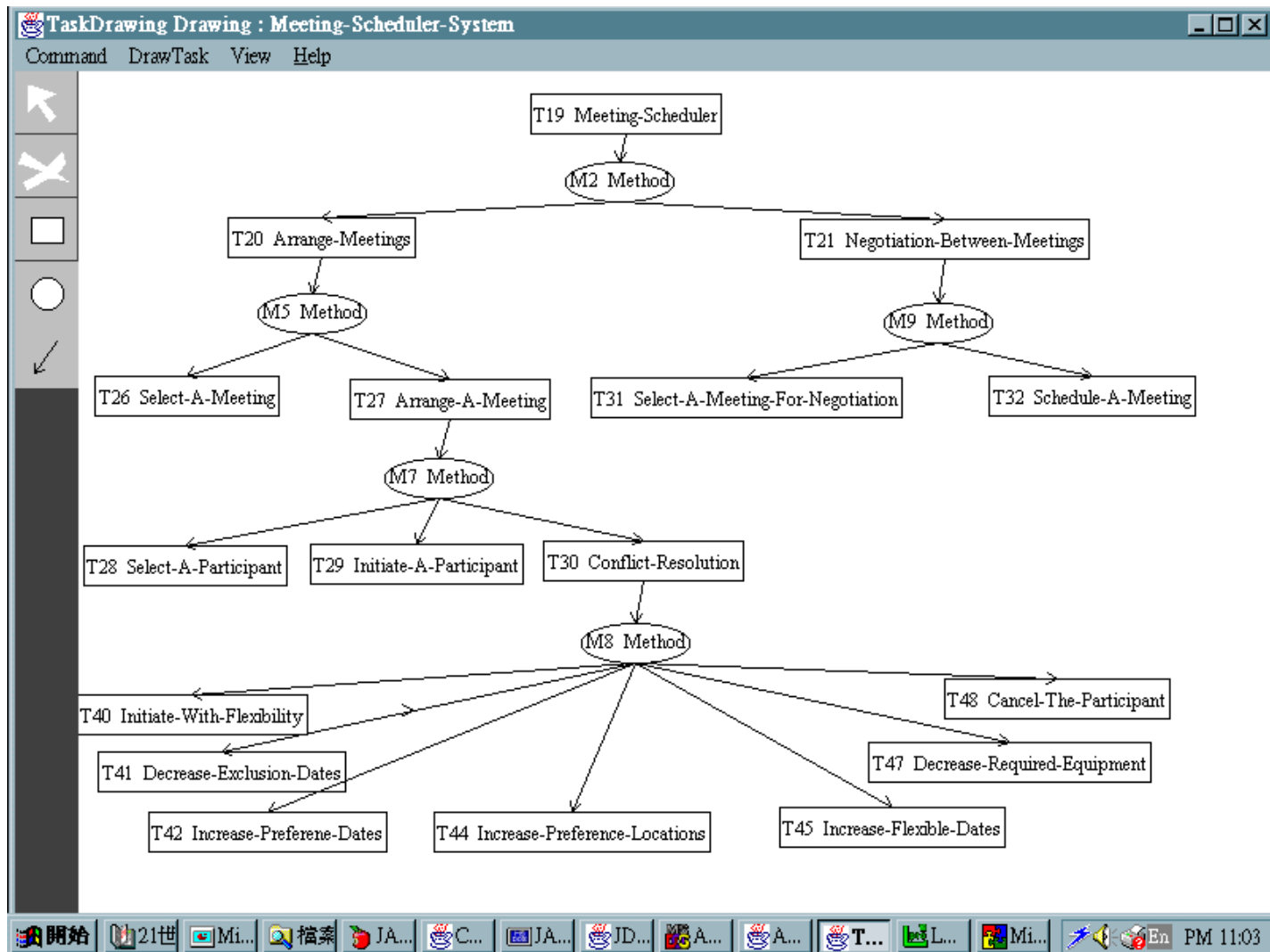
- 軟體程式自動測試

- 例子:

- MicroStep、Rational ROSE、Borland Together、TBCG



會議排程系統的 TBCG 軟體規格



自動產生會議排程系統的輸入畫面

title

File Add Component

Input Meeting Data

Input Location Data

Input Participant

Execute MS System

Exit

Input Loc...

File Add Component

id E1-12

Equipment Projector

Save

Input Meeting Data

File Add Component

id se01

Significance 5

Begin-Range 0420

End-Range 0510

Time-Period 4

Participant i03 i04 i05 i06 i

Participant-Importance 2 3 8 6 10 4 5

Available-Date 0420 0421 0422 0423

Available-Location

Meeting-Date E1-124 E1-105 E1-11

Meeting-Location

Save

Input Participant

File Add Component

id i06

Status 9

Exclusion-Date 0424 0425 0430 0501

Preference-Date 0420 0421 0427 0428 05

Preference-Location E1-124 E1-105 E1-220

Required-Equipment projector microphone

Flexible-Date 423 424 425 426 427

Save



會議排程系統的 排程結果

Meeting :

- se04
- se03
- se01
- se02

Participant :

- i01
- i09
- i10
- i02

Location :

- E1-115
- E1-120
- E1-220
- E1-108

Exit

Meeting Data

Id	se04
Significance	3
Begin Range	428
End Range	505
Time Period	3
Participant	i01 i09 i10
Importance	10 9 8
Date	429
Meeting Date	430 501 428
Location	E1-124 E1-120
Meeting-Location	E1-220

Previous Next Close

Location Data

Id	E1-115
Equipment	projector internet

Previous Next Close

Participant Data

Id	i01
Status:	10
Exclusion Date:	423 424 425 430 501
Preference Date	420 421 427 428 503 504 505 506 509 510
Location	E1-105 E1-108 E1-220 E1-120 E1-115
Equipment	projector microphone
Attend Meeting	se04
Weight	10
Flexible date	423 424 425 426 427 428 429 430 501

Previous Next Close



結論

- 軟體工程的終極目標？
 - 自動化程式設計
- 加入人工智慧是否有可能達成目標？
 - Specific vs General
- 電腦是否可能有自行思考之能力？
 - Strong AI vs Weak AI

