**物件導向程式設計作業**

|  |
| --- |
| **命題者︰Zhou Yuan** |
| **題目名稱(中文/英文)：****Payroll System** |
| **主要測試觀念：**   |  |  | | --- | --- | | **Basics** | **Functions** | | ■ C++ BASICS  ■ FLOW OF CONTROL  ■ FUNCTION BASICS  □ PARAMETERS AND OVERLOADING  ■ ARRAYS  □ STRUCTURES AND CLASSES  □ CONSTRUCTORS AND OTHER TOOLS  □ OPERATOR OVERLOADING, FRIENDS, AND REFERENCES  □ STRINGS  ■ POINTERS AND DYNAMIC ARRAYS | □ SEPARATE COMPILATION AND NAMESPACES  ■ STREAMS AND FILE I/O  □ RECURSION  □ INHERITANCE  ■ POLYMORPHISM AND VIRTUAL FUNCTIONS  □ TEMPLATES  □ LINKED DATA STRUCTURES  □ EXCEPTION HANDLING  ■ STANDARD TEMPLATE LIBRARY  □ PATTERNS AND UML | |
| **題目說明：**  A company pays its employees weekly. The employees are of three types: Salaried employees are paid a fixed weekly salary regardless of the number of hours worked, commission employees are paid a percentage of their sales and base-salary-plus-commission employees receive a base salary plus a percentage of their sales. The company wants to implement a C++ program that performs its payroll calculations polymorphically.  The following table is the fields of single employee.   |  |  | | --- | --- | | Name | Type | | ID | string | | first\_name | string | | last\_name | string | | sales | unsigned int | | salary | double |   The system should be including the following functions:   1. Insert employee: insert a new employee into the system, record the employee’s **type, ID**, **first\_name** and **last\_name**.  * If type = 0, he/she is a salaried employee, it should also record the **fix\_salary**. * If type=1, he/she is a commission employee, it should also record the **commission\_rate**. * If type=2, he/she is a base-salary-plus-commission employee, it should also record the **commission\_rate** and **base\_salary**.  1. Record **ID sales**: record the employee’s sales whose ID matches the given ID.  * Update the related fields. * If the id is not in the system, please print “The employee **ID** is not found”. and break;  1. Update **ID** fields: update the employee whose ID matches the given ID.  * If the field name isn’t exist, please print “Employee **ID** has no field XX ” and break; * If the id is not in the system, please print “The employee **ID** is not found”.  1. Delete **ID**: delete the employee whose **ID** matches the given **ID**.  * If the **ID** is not in the system, please print “The employee **ID** is not found” and break;  1. List: list each attribute value of all employees ordered by **salary**.  * Ouput format:  **ID first\_name last\_name sales salary** (**fix\_salary**/**base\_salary/commission\_rate**)  1. Exit: exit the program.   **輸入說明：See the Sample Input.**  **輸出說明：See the Sample Output.**  **IO範例 :**   |  |  |  | | --- | --- | --- | |  | **Sample Input** | **Sample Output** | | 第一組 | Insert 0 A01 Amy White 10000  Insert 0 A02 John Black 15000  Insert 1 B01 Cindy White 0.6  Insert 1 B02 Mike White 0.2  Insert 2 C01 Andrew Smith 8000 0.4  Record A01 4500  Record A02 3500  Record A01 2500  Record B01 5500  Record C01 4500  Update A01 fix\_salary=8000 commission\_rate=0.6  Update A03 fix\_salary=8000  Delete B02  Update B02 last\_name=AAA  List | Employee A01 has no field commission\_rate  The employee A03 is not found  The employee B02 is not found  A02 John Black 3500 15000 15000  C01 Andrew Smith 4500 9800 8000 0.4  A01 Amy White 7000 8000 8000  B01 Cindy White 5500 3300 0.6 | |
| * + 易，僅需用到基礎程式設計語法與結構 * 中，需用到多項程式設計語法與結構 * 難，需用到多項程式結構或較為複雜之資料型態或結構 |
| **解題時間：60**分鐘。 |
| **其他註記：** |