**CPP Problem Design**

|  |
| --- |
| **Subject:** **Hot dog stand** |
| **Contributor: 林承達, 陳俊儒, 廖宣瑋** |
| **Main testing concept:** Class   |  |  | | --- | --- | | **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 | |
| **Description:**  Your boss has a hot dog chain in the city, and he needs your help to manage his hot dog stores.   * Please define a class called **HotDogStand()** to store and manage the hot dog stores. * The class HotDogStand has three variables: * **char \*standId**：The ID of the store. * **int hotDogSellAmount**：The hot dog sales volume of the store. * **static int totalSellAmount**：The total hot dog sales volume of all stores. * The class HotDogStand has two constructors: * **HotDogStand (id)**: Construct a HotDogStand with the given ID and set the sales volume to 0. * **HotDogStand (id, amout)** : Construct a HotDogStand with the given ID and sales volume. * You are required to implement the following member functions: * **justSold()**: Increase the hot dog sales volume by 1. * **print()**：Print the ID and sales volume of the store separated by spaces. * **thisStandSoldAmount()**：Return the sales volume of the store. * **allStandSoldAmount()**：Return the total sales volume of all stores.   **Input:**  The **main()** function in your submission will be replaced when judging.  You can use the **main()** function in “**Other Notes**” to test your program.  This exercise does not have an input.  **Output:**  The result of executing your program with the given main function.  **Sample Input / Output：**   |  |  | | --- | --- | | Sample Input | Sample Output | | No inputs | Stand1 1  Stand2 101  Stand3 1  Total Sold : 103 | |
| **■ Eazy,Only basic programming syntax and structure are required.**  **□ Medium,Multiple programming grammars and structures are required.**  **□ Hard,Need to use multiple program structures or more complex data types.** |
| **Expected solving time:**  15 minutes |
| **Other notes:**  int main(void)  {  HotDogStand stand1("Stand1", 0);  HotDogStand stand2("Stand2", 100);  HotDogStand stand3("Stand3");  stand1.justSold();  stand2.justSold();  stand3.justSold();  stand1.print();  stand2.print();  stand3.print();  std::cout << "Total Sold : " << HotDogStand::allStandSoldAmount  () << std::endl;  return 0;  } |