**COVER PAGE**

REPLACE WITH BLUE COVER

PROVIDED

**PROJECT EVALUATION**

|  |  |  |  |
| --- | --- | --- | --- |
| **GROUP MEMBERS** | | **STUDENT NUMBER** | |
| 1. | |  | |
| 2. | |  | |
| 3. | |  | |
| 4. | |  | |
|  | |  | |
| **CONTENTS** | | | **MARKS** |
| 1. **Synopsis of the system** | * describe all the input data needed (2 marks) * describe the process on how program works (2 marks) * describe all the information (output) to be displayed (2 marks) | | /06 |
| 2. **Flow Chart** | Diagrams - use correct diagrams | | /04 |
| Flow of Program - show logic flow | | /04 |
| Explanation - use words to explain the action | | /04 |
| Input - list all the variables needed | | /04 |
| Process - state all the process needed | | /04 |
| Output - list all the variables needed | | /04 |
| **CO3:LO2** | | | **/30** |
| 3. **Source Code** | Correctness - The system can solve the assigned problem and output should be as per the specifications. | | /05 |
| Reliability - The system should function accurately for a long period of time and also function correctly over all ranges and combination of data. | | /05 |
| User friendliness - Easy to use with enough information. | | /05 |
| Efficiency - Use of techniques covered and provides quality output.  - sequential (5 marks)  - selection (8marks)  - repetition (7 marks)  - function (5 marks)  - input /output statements (5marks) | | /30 |
| Readability of code - The source code should be simple and easy to understand. | | /05 |
| 4. **Screen output printed**   * menu (2 marks) * layout creativity (3 marks) | | | /05 |
| 5. **Complexity of a program** | | | /10 |
| 6. **Overall effort in the project** | | | /05 |
| **CO4:LO6** | | | **/70** |
| **TOTAL** | | | **/100** |

**TOTAL MARKS = / 100**

**TABLE OF CONTENTS**

**1.0 SYSTEM SYNOPSIS**

* describe all the input data needed
* describe the process on how program works
* describe all the information (output) to be displayed

**2.0 ANALYSIS AND FLOWCHART**

**Function menu ()**

|  |  |
| --- | --- |
| **Input:** | - |
| **Process:** | 1. BEGIN 2. Display code and destination 3. END |
| **Output:** | Display code and destination menu |

**Function menu ()**

2.

**Function calcPrice (int code,int quantity)**

|  |  |
| --- | --- |
| **Input:** | code and quantity |
| **Process:** | EXAMPLE  BEGIN  DECLARE double ticketPrice, totalPrice, insurance;  string destination;  If code = 1  Assign ticketPrice = 45;  Assign destination = “Kuala Lumpur”;  else if code = 2  Assign ticketPrice = 28  Assign destination = “Ipoh”;  else if code = 3  Assign ticketPrice = 55  Assign destination = “Kuala Terengganu”;  else  Assign ticketPrice = 0  Assign destination = “Invalid destination code”;  CALCULATE insurance = ticketPrice\*0.05;  CALCULATE totalPrice = (ticketPrice + insurance) \* quantity;  DISPLAY code, destination, quantity, ticketPrice, insurance, totalPrice  END |
| **Output:** | code, destination, quantity,ticketPrice,insurance,totalPrice |

**Function calcPrice (int code,int quantity)**

ii.

i.

3.

2.

4.

i.

ii.

ii.

5.

i.

6.

i.

ii.

7.

8.

9.

**Function main()**

|  |  |
| --- | --- |
| **Input:** | code, quantity, nextCustomer |
| **Process:** | 1. BEGIN 2. DECLARE int code, quantity;   char nextCustomer;   1. REPEAT if answer is yes    1. Call function menu()    2. Input code, quantity    3. Call function calcPrice(code,quantity)    4. Input repeat next customer 2. DISPLAY “thank you” message 3. END |
| **Output:** | - |

**Function main()**

2.

3.

**yes**

b.

a.

c.

**no**

4.

d.

**3.0 SOURCE CODE**

#include <iostream>

using namespace std;

void menu(); //function prototype

void calcPrice(int,int); //function prototype

void menu() //function definition

{

}

void calcPrice(int code,int quantity) //function definition

{

}

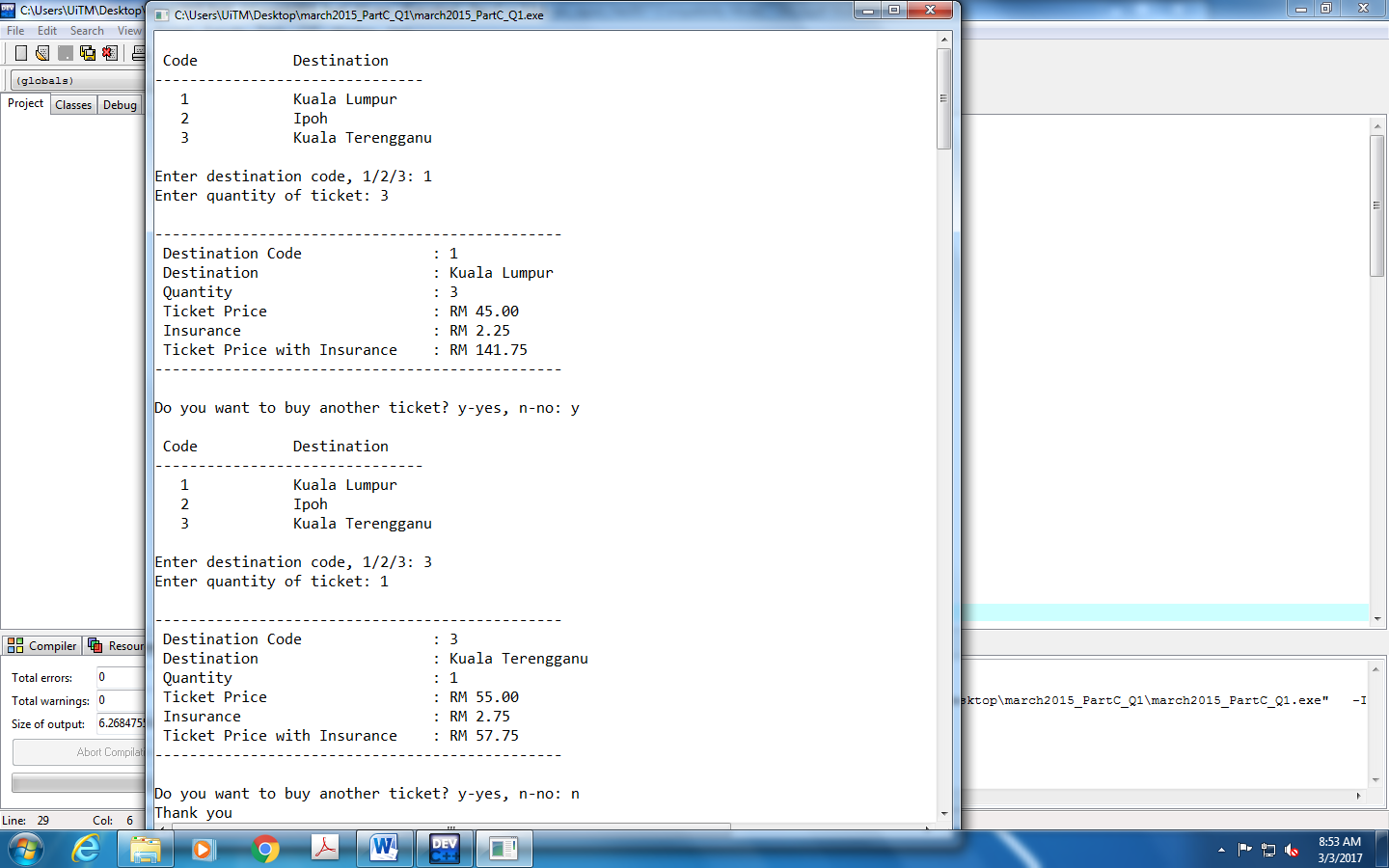
int main()

{

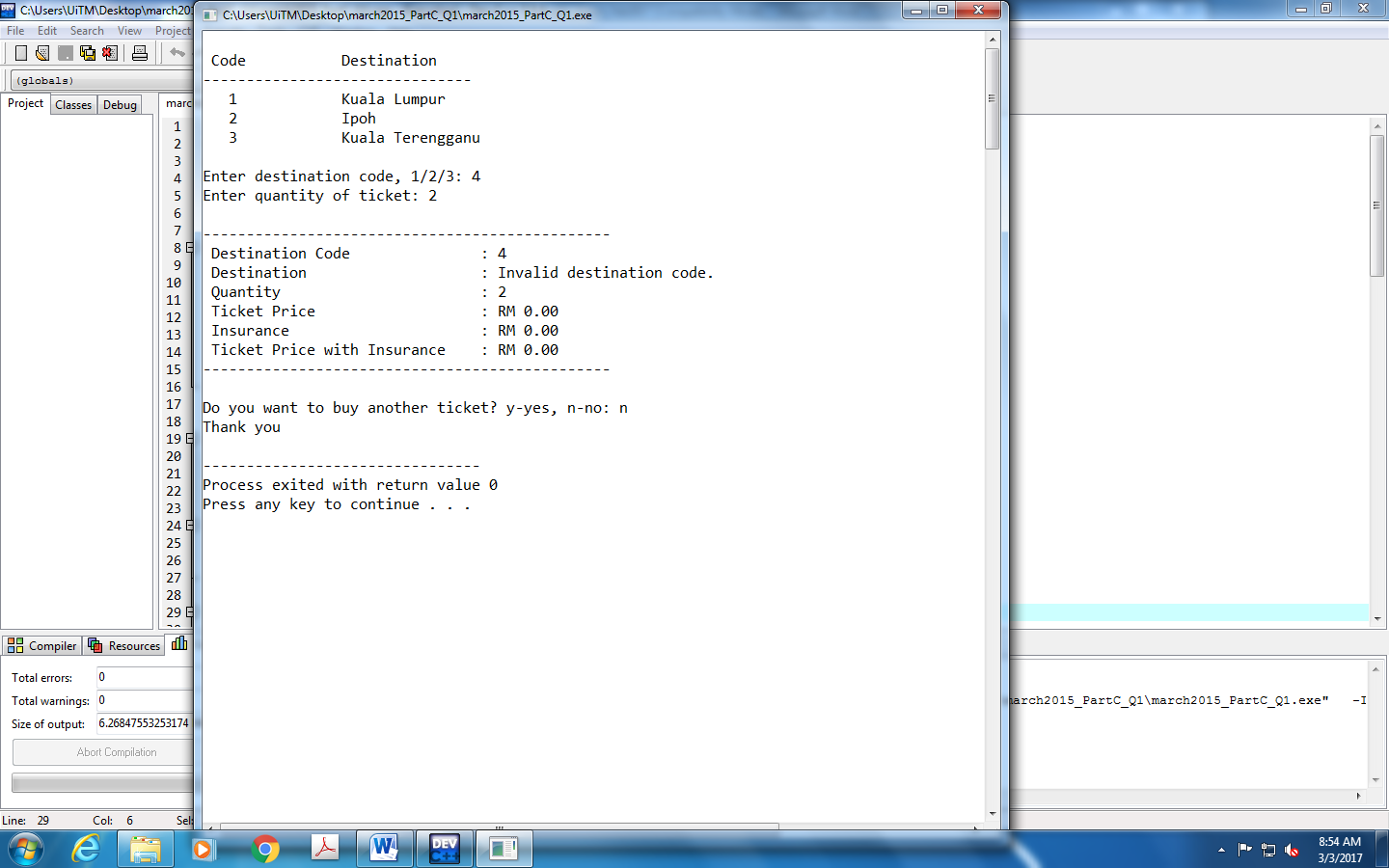
}

**4.0 SAMPLE OUTPUTS**

Sample Output 1



Sample Output 2



**CD/DVD**

**in an Envelope**