**CIS3100**

**KARY HUANG**

**ASSIGNMENT #2**

**03/09/2016**

**Pseudocode with stepwise refinement**

1. Ask the user to enter their employee ID, previous salary and current salary; assign to *empl\_id, prev\_salary* and *current\_salary* respectively.
2. Define the difference between current salary and previous salary as *salary\_increase* and print along with employee ID.
3. If salary increase is greater than $5000, print “nice increase”; if salary increase is between $1000 and $5000 inclusively, print “fair increase”; if salary increase is less than $1000, print “small increase”.
4. Give *max\_increase* an initial value of 0.
5. If *salary\_increase* is greater than *max\_increase*, replace its value; then assign the corresponding *empl\_id* to *max\_employee*.
6. Repeat procedures 1-3 and 5 for ten times.
7. Print *max\_increase* and *max\_employee*.

**INPUT**

//Kary Huang Assignment 2

#include <iostream>

#include <cmath>

using namespace std;

int main()

{

int empl\_id, current\_salary, salary\_increase;

int max\_increase, max\_employee, prev\_salary;

max\_increase = 0;

int x;

for (int x = 1; x <= 10; x++)

{

cout << "Please enter your four digit employee ID: " << endl;

cin >> empl\_id;

cout << "Please enter last year's salary: " << endl;

cin >> prev\_salary;

cout << "Please enter this year's salary: " << endl;

cin >> current\_salary;

salary\_increase = current\_salary - prev\_salary;

cout << "For employee ID " << empl\_id << ',' << " salary has increased by $" << salary\_increase << ", ";

if (salary\_increase > 5000)

cout << "nice increase!" << endl << endl;

else if (salary\_increase >= 1000 && salary\_increase <= 5000)

cout << "fair increase!" << endl << endl;

else if (salary\_increase < 1000)

cout << "small increase!" << endl << endl;

if (salary\_increase > max\_increase)

{

max\_increase = salary\_increase;

max\_employee = empl\_id;

}

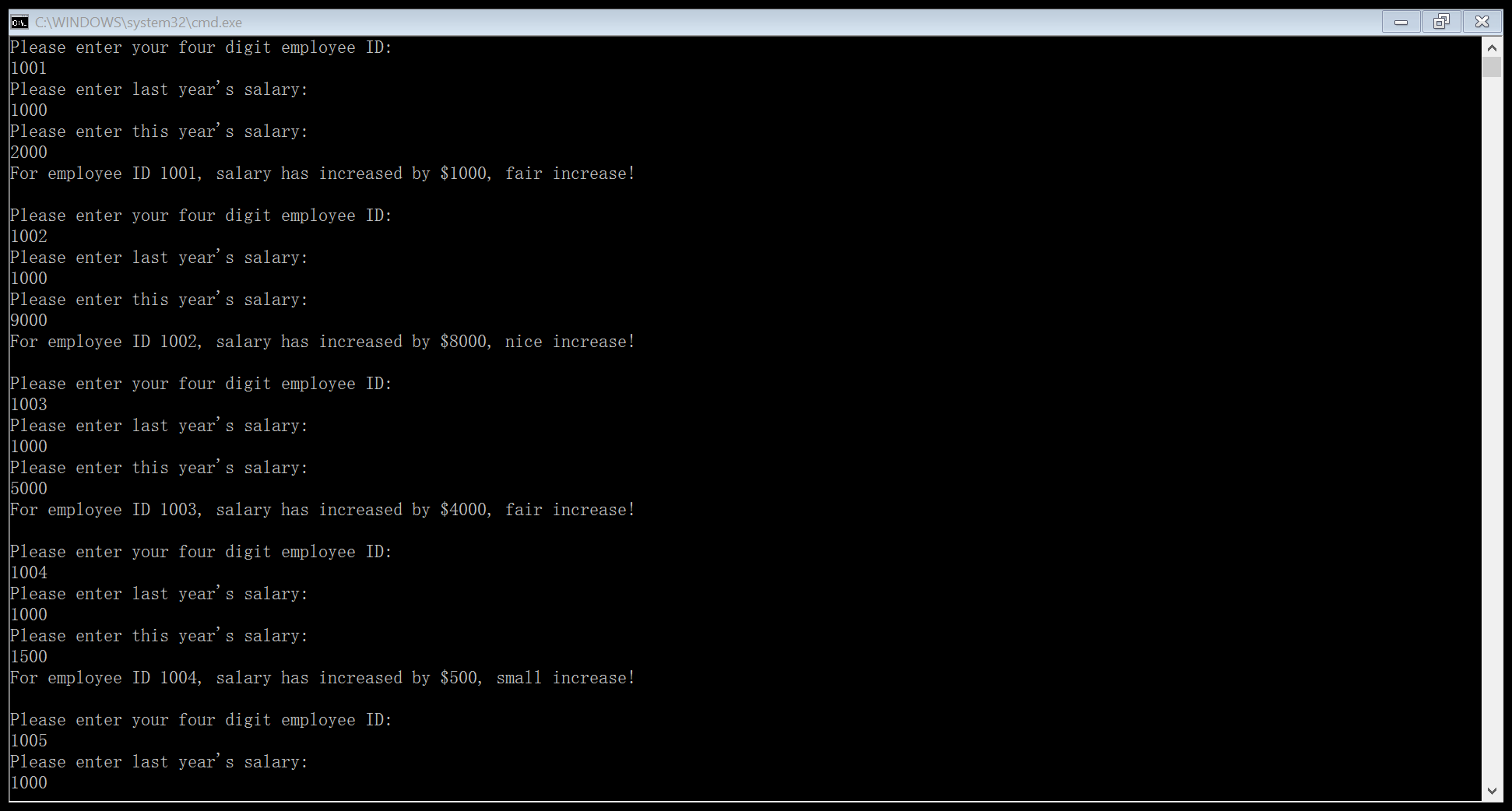
}

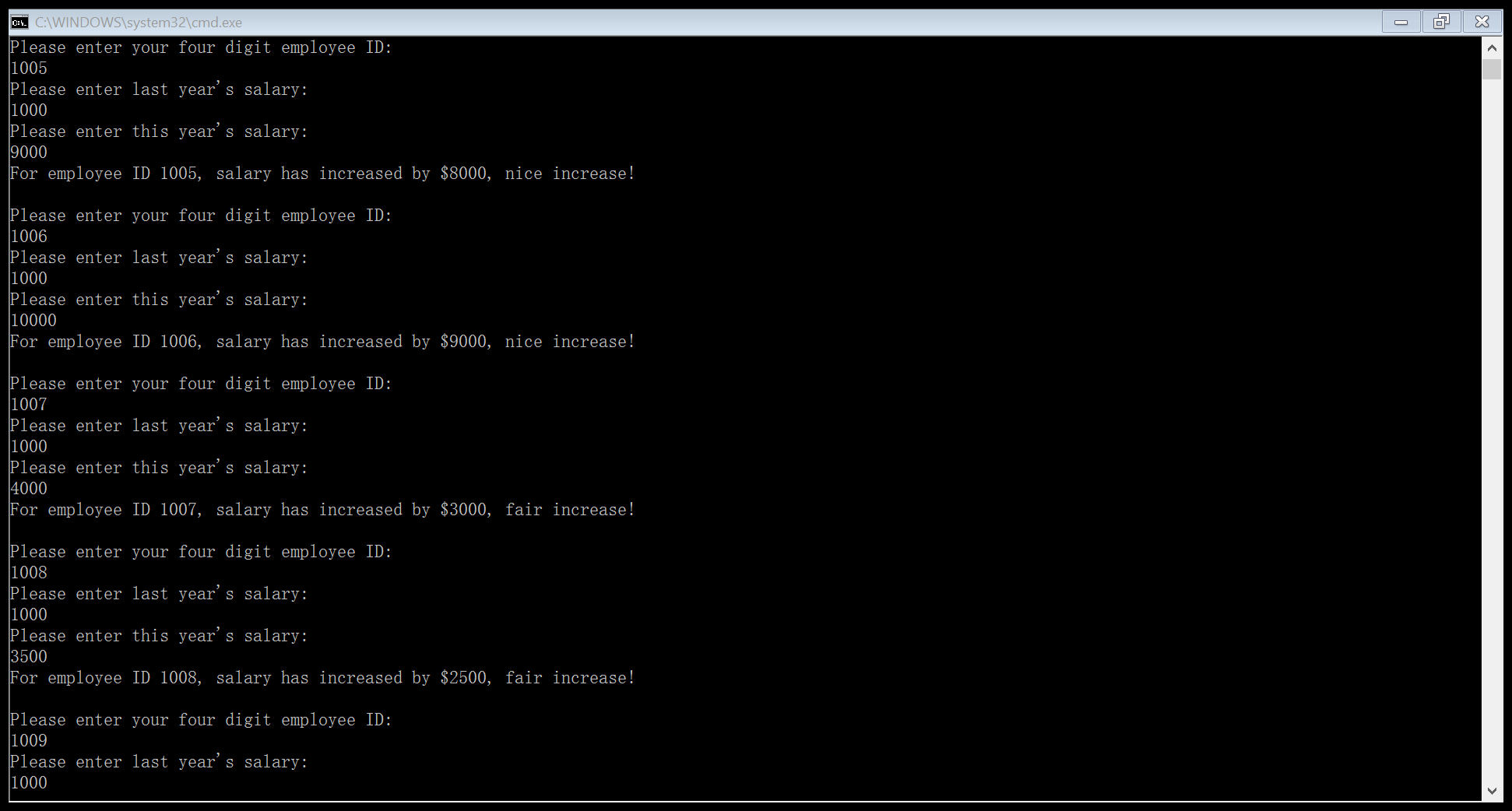
cout << "Employee with ID " << max\_employee << " has the maximum increase in salary which is $" << max\_increase << '.' << endl << endl;

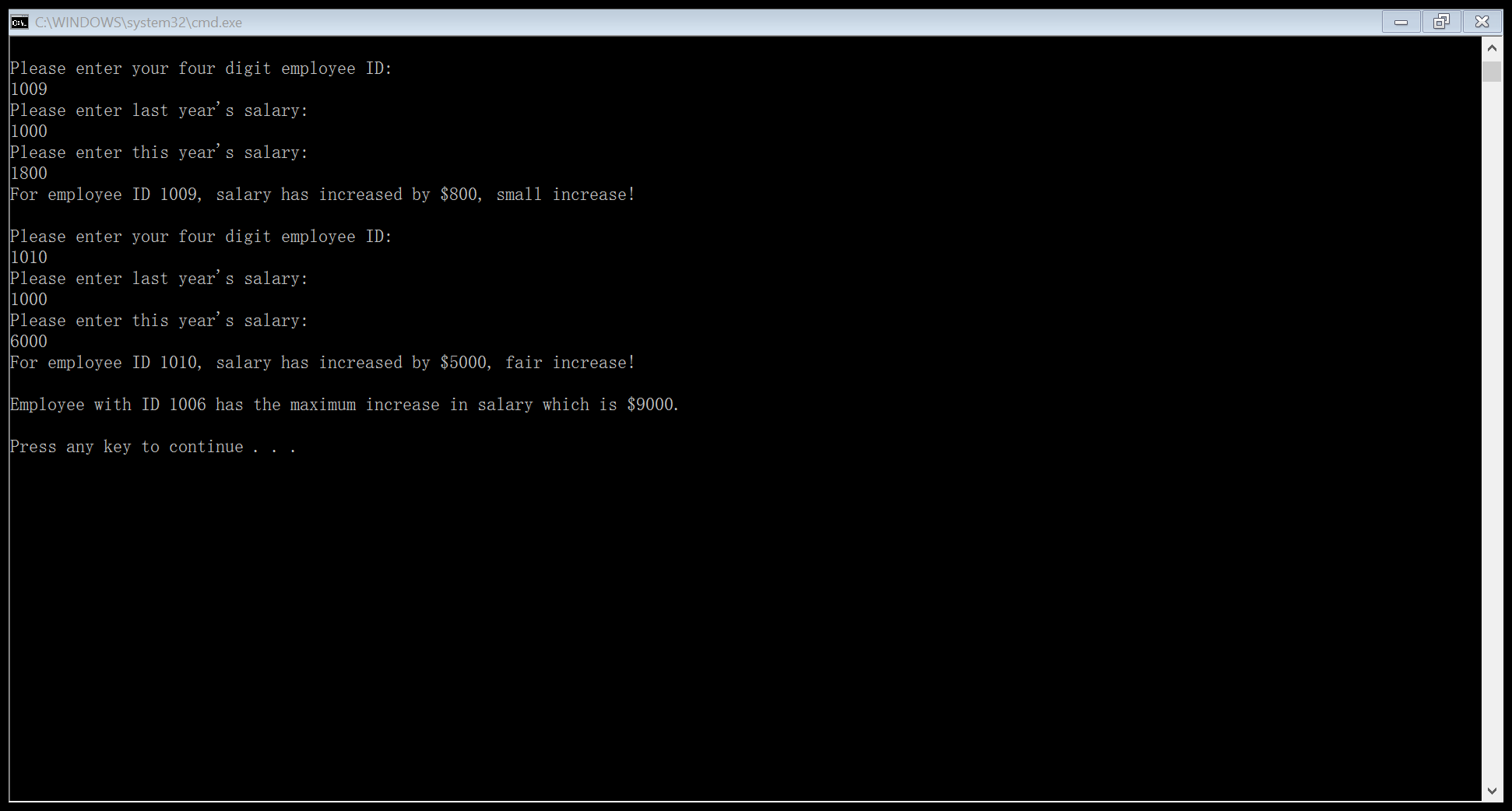
return 0;

}

**OUTPUT**

****

****

****