**C++ Week 11 E-Tutorial**

**1(a)**

BankAcct::BankAcct(int i, string s, double d) {

ofstream out\_file;

out\_file.open("IN.dat", ofstream::app);

out\_file << i << " " << s << " " << d << endl;

out\_file.close();

}

void BankAcct::writeFile(int t) {

ifstream in\_file;

ofstream out\_file;

in\_file.open("IN.dat");

out\_file.open("OUT.dat");

double total = 0;

while (in\_file >> type >> acctNum >> balance) {

if (type == t) {

out\_file << type << setw(8) << acctNum << setw(7) << balance << endl;

total += balance;

}

}

out\_file << "Total : " << setw(8) << total << endl;

in\_file.close();

out\_file.close();

}

**1(b)**

#include "stdafx.h"

#include "BankAcct.h"

int main()

{

BankAcct test;

BankAcct newBankAcct(1, "B98765", 20000);

int type;

cout << "Enter the Bank Account Type Number : ";

while (cin >> type) {

if (type == 1 || type == 2) {

break;

}

else {

cout << "Wrong Account Type Number!" << endl;

cout << "Enter the Bank Account Type Number : ";

}

}

test.writeFile(type);

system("PAUSE");

return 0;

}

**2(a)(i)**

class Employee

{

public:

Employee();

virtual void setValue(string, string, double);

virtual void displayValue();

string getName();

~Employee();

private :

string empID;

string empName;

double salary;

};

Employee::Employee()

{

empID = "NIL";

empName = "NotAnEmployee";

salary = 0;

}

void Employee::setValue(string id, string name, double sal) {

empID = id;

empName = name;

salary = sal;

}

void Employee::displayValue() {

cout << "Basic Details : " << empID << " " << empName << endl;

}

string Employee::getName() {

return empName;

}

**2(a)(ii)**

class Regular :

public Employee

{

public:

Regular();

void setValue(string, string, double, int);

void displayValue();

~Regular();

private :

int annualLeave;

};

Regular::Regular() : Employee()

{

annualLeave = 0;

}

void Regular::setValue(string id, string name, double sal, int annual) {

Employee::setValue(id, name, sal);

annualLeave = annual;

}

void Regular::displayValue() {

Employee::displayValue();

cout << "Annual Leave for regular employee " << Employee::getName() << " : " << annualLeave << endl;

}

**2(a)(iii)**

class Casual :

public Employee

{

public:

Casual();

void setValue(string, string, double);

void displayValue();

~Casual();

};

Casual::Casual() : Employee()

{

}

void Casual::setValue(string id, string name, double sal) {

Employee::setValue(id, name, sal);

}

void Casual::displayValue() {

Employee::displayValue();

cout << Employee::getName() << " is a casual employee" << endl;

}

**2(b)**

vector<Employee\*> Employees;

Employee\* Employee1 = new Employee;

Regular\* Regular1 = new Regular;

Casual\* Casual1 = new Casual;

Employee1->setValue("S001", "Betty Tan", 1000);

Regular1->setValue("S002", "Peter Lim", 1000, 14);

Casual1->setValue("S003", "Rebecca Lim-Tan", 1000);

Employees.push\_back(Employee1);

Employees.push\_back(Regular1);

Employees.push\_back(Casual1);

for (int i = 0; i < Employees.size(); i++) {

Employees[i]->displayValue();

}