SECJ 1013 PROGRAMMING TECHNIQUE 1

ASSIGNMENT 3

GROUP MEMBER'S (MATRIC NO):

SECTION:

1) State whether the following declarations are valid or invalid. Give reasons for the invalid declarations and draw memory layout for the valid declarations. **(7 marks)**

```
 i. int var = 25;
    int *ptr = &var;
ii. int var = 30;
    int* ptr = var;
iii. int var, *ptr;
    ptr = &var;
iv. float fvar;
    int *ptr = &fvar;
 v. float fvar, *fptr = &fvar;
vi. int *ptr = &var;
    int var = 25;
vii. double* dptr1, dptr2;
    double dvar = 25.2;
    dptr1 = &dvar;
    dptr2 = &dvar;
```

2) Determine the output and draw a memory layout (or memory allocation) of the pointers and variables for code segment below. Note: Draw a memory layout that represents C++ statement line by line. **(7 marks)**

```
int  x = 10, y = 20, z = 30;
int *ptr;

cout << x << " "  << y << " " << z << endl;
ptr = &x;
*ptr *= 10;
ptr = &y;
*ptr *= 4;
ptr = &z;
*ptr *= 2;

cout << x << " "  << y << " " << z << endl;
```

3) Write two statements to free dynamically allocated array and double which are declared as follows: **(2 marks)**

```
int *iPtr = new int [100];
double *dPtr = new double;
```

4) Starting address of the following array named iVar is 0xFEC07.

[0]	[1]	[2]	[3]
2	5	8	6

iVar

What is the output that will be displayed based on the following statements? **(4 marks)**

i. cout << iVar; = 0x7afe40
ii. cout << iVar [0]; = 2
iii. cout << *iVar; = 2
iv. cout << *(iVar + 2); = 8

5) Write a structure declaration to hold the following data **(6 marks)**

i. About a flight reservation: passenger name, age, reservation code, departure location, destination, flight number, departure time, arrival time, cost and payment status.

```
#include <iostream>
using namespace std;

const int SIZE = 25;
const int CODE = 8;

struct Flight_Reservation
{
    char Passenger_Name[SIZE];      // Passenger's Name
    int age;                        // Passenger's Age
    char Reserve_Code[CODE];        // Passenger's Reservation Code
    char Depart_Location[SIZE];     // Passenger's Departure Location
    char Destination[SIZE];         // Passenger's Destination
    char Flight_Number[CODE];       // Passenger's Flight Number
    int Depart_Time;                // Passenger's Departure Time
    int Arrive_Time;                // Passenger's Arrival Time
    double Cost;                    // Passenger's Flight Cost
    bool Payment_Status;            // Passenger's Payment Status
};
```

ii. About saving account: account number, account balance, interest rate, total deposit and total withdraw.

```
#include <iostream>
using namespace std;

const int AccNo = 18;

struct SavingAcc
{
    int Account_Number [AccNo];     // Account Number
    double Account_Balance;         // Account Balance
    double Interest_Rate;           // Interest Rate
    double Total_Deposit;           // Total Deposit
    double Total_Withdraw;          // Total Withdraw
};
```

iii. About PT1 assessments: student's name, test 1, assignment, quiz, lab exercise, final exam, course work mark, total mark and grade.

```cpp
#include <iostream>
using namespace std;

const int SIZE = 25;

struct PT1ASSESSMENTS
{
    char name[SIZE];     // Student's name
    int test1;           // test 1 mark
    int assignment;      // assignment
    int quiz;            // quiz
    int lab_exercise;    // lab exercise
    int final_exam;      // final exam
    int course_work;     //course work mark
    int total_mark;      // total mark
    char grade;          // grade
};
```

6) A car salesman keeps the information of each model of car he sells. The example of information for 3 cars' models is as in Table 2. Write C++ statement for the following task. **(10 marks)**

Model	Engine capacity	Price
Waja	1.6	60000
Wira	1.5	50000
MyVi	1.3	45000

i. Define a structure for storing the above information named Car.

ii. Declare a variable called myCar and initialized it with some values of your choice. Display information on myCar.

iii. Declare another variable called mySecondCar and assign values to it using assignment statements. Display information on mySecondCar.

iv. Print the total of price paid for myCar and mySecondCar.

v. Copy the values and information of mySecondCar into myCar and display current information on myCar.

```cpp

#include <iostream>
#include <string>
using namespace std;

struct Car
{
    string Model;
    double Engine_Capacity;
    double Price;
};

int main()
{
    Car myCar = {"Persona", 1.9, 60000 };
    cout << "Here is information on my car: " << endl;
    cout << "Model: " << myCar.Model << endl;
    cout << "Engine capacity of " << myCar.Model << "is: " << myCar.Engine_Capacity << endl;
    cout << "The price of " << myCar.Model << "is: " << myCar.Price << endl;

    Car mySecondCar = { "Myvi", 1.5, 50000};
    cout << "\nHere is information on my second car: " << endl;
    cout << "Model: " << mySecondCar.Model << endl;
    cout << "Engine capacity of " << mySecondCar.Model << "is: "<< mySecondCar.Engine_Capacity << endl;
    cout << "The price of " << mySecondCar.Model << "is: " << mySecondCar.Price << endl;

    double TotalPrice = myCar.Price + mySecondCar.Price;
    cout << "Total price of both car is: " << TotalPrice << endl;

    myCar = mySecondCar;
    cout << "\nUpdated myCar informations: " << endl;
    cout << "Model: " << myCar.Model << endl;
    cout << "Engine capacity of " << myCar.Model << "is: " << myCar.Engine_Capacity << endl;
    cout << "The price of " << myCar.Model << "is: " << myCar.Price << endl;

    return 0;
}
```

7) Write the code segment for each of the following tasks: **(8 marks)**

a) Declare a structure type:

i. named `Salary`, with the following members:

`basic` : a double value

`allowances` : a double value

```
struct Salary {
 double basic;
 double allowance;
};
```

ii. named `Employee`, with the following members:

`id` : an integer value

`salary` : a `Salary` structure variable

```
struct Employee {
 string name;
 int id;
 Salary salary;
};
```

iii. Declare a variable of structure type `Employee` named `myEmp`.

- `Employee myEmp = { "Azira", 8902, {4500.0, 500.0}};`

b) By using the variables and structure declaration in (a), define a function named `displayEmp`. It should accept an `Employee` structure variable as its argument and not return a value. The function should display the contents of the variable onto the screen based on figure below. *Notes: Assuming the data for struct members was already assigned.

```
Sample output:
Name: Azira
Id: 8902
Basic salary: RM 4500
Allowances: RM 500
```

```
void displayEmp ( Employee myEmp ){

        cout << "Name: " << myEmp.name << endl;
        cout << "Id: " << myEmp.id << endl;
        cout << "Basic Salary: " << myEmp.salary.basic << endl;
        cout << "Allowances: " << myEmp.salary.allowance << endl;
```