EEE101 C Programming and Software Engineering
Solutions to Lab Practice 11

Exercise 1

```c
#include<stdio.h>

struct personal{                                    /*define structure personal*/
        char name[20];
        int day;
        char month[20];
        int year;
        float salary;
        };

int main(){

printf("Input Values\n");
scanf("%s  %d  %s  %d  %f", person.name, &person.day, person.month, &person.year,
&person.salary);
printf("%s %d %s %d %f\n", person.name, person.day, person.month, person.year,
person.salary);
}
```

Exercise 2

```c
#include<stdio.h>
#define NUMRECS 5

struct PayRecord{                               /construct a global structure type*/
        int id;
        char name[20];
        double rate;
        };

int main(){
int i;
struct PayRecord employee[NUMRECS] = {          /*define an array of structures*/
        {123,"Tom",12.3},
        {124,"Paul",14.3},
        {125,"Mary",12.39},
        {126,"John",22.3},
        {128,"Susan",9.12}
        };
for (i=0; i<NUMRECS; i++)
        printf("%d  %-20s  %4.2f\n",employee[i].id,  employee[i].name,  employee[i].rate);
return 0;
}
```

Exercise 3

It is a very simple program which calculates the net pay (through passing a structure as argument to function).

Exercise 4

```c
#include<stdio.h>

struct Employee{
        int idNum;
        double payRate;
        double hours;
        };

double calcNet(struct Employee *);

int main(){
struct Employee emp = {6768,  8.93,  40.5};
double netpay;

netpay = calcNet(&emp);

printf("The  net pay  of  employee  %d  is RMB  %6.2f\n", emp.idNum,netpay);

return 0;
}

double calcNet(struct Employee *pt){
        return(pt->payRate*pt->hours);
        }
```