Week 6 Lab: Structures – Suggested Solutions

Q1: (computeExp)

```c
#include <stdio.h>
typedef struct {
   float operand1, operand2;
   char op;
} bexpression;
float compute1(bexpression expr);
float compute2(bexpression *expr);
int main()
{
   bexpression e;
   int choice;

   printf("Select one of the following options: \n");
   printf("1: compute1()\n");
   printf("2: compute2()\n");
   printf("3: exit()\n");
   do {
      printf("Enter your choice: \n");
      scanf("%d", &choice);
      switch (choice) {
         case 1:
            printf("Enter expression (op1 op2 op): \n");
            scanf("%f %f %c", &e.operand1, &e.operand2, &e.op);
            printf("compute1(): %.2f\n", compute1(e));
            break;
         case 2:
            printf("Enter expression (op1 op2 op): \n");
            scanf("%f %f %c", &e.operand1, &e.operand2, &e.op);
            printf("compute2(): %.2f\n", compute2(&e));
            break;
      }
   } while (choice < 3);
   return 0;
}
float compute1(bexpression expr)
{
   float result;

   switch (expr.op) {
      case '+': result = expr.operand1 + expr.operand2;
         break;
      case '-': result = expr.operand1 - expr.operand2;
         break;
      case '*': result = expr.operand1 * expr.operand2;
         break;
      case '/': result = expr.operand1 / expr.operand2;
         break;
   }
   return result;
}
float compute2(bexpression *expr)
{
   float result;
   switch (expr->op) {
      case '+': result = expr->operand1 + expr->operand2;
         break;
      case '-': result = expr->operand1 - expr->operand2;
```

```
            break;
         case '*': result = expr->operand1 * expr->operand2;
            break;
         case '/': result = expr->operand1 / expr->operand2;
            break;
      }
      return result;
}
```

Q2: (phoneBook)

```
#include <stdio.h>
#include <string.h>
#define MAX 100
typedef struct {
   char name[20];
   char telno[20];
} PhoneBk;
void printPB(PhoneBk *pb, int size);
int readin(PhoneBk *pb);
void search(PhoneBk *pb, int size, char *target);
int main()
{
   PhoneBk s[MAX];
   char t[20], *p;
   int size=0, choice, dummychar;

   printf("Select one of the following options: \n");
   printf("1: readin()\n");
   printf("2: search()\n");
   printf("3: printPB()\n");
   printf("4: exit()\n");
   do {
      printf("Enter your choice: \n");
      scanf("%d", &choice);
      switch (choice) {
         case 1:
            scanf("%c", &dummychar);
            size = readin(s);
            break;
         case 2:
            scanf("%c", &dummychar);
            printf("Enter search name: \n");
            fgets(t, 20, stdin);
            if (p=strchr(t,'\n')) *p = '\0';
            search(s,size,t);
            break;
         case 3:
            printPB(s, size);
            break;
      }
   } while (choice < 4);
   return 0;
}
void printPB(PhoneBk *pb, int size)
{
   int i;

   printf("The phonebook list: \n");
   if (size==0)
      printf("Empty phonebook\n");
   else {
```

```
      for (i=0; i<size; i++) {
         printf("Name: %s\n", (pb+i)->name);
         printf("Telno: %s\n", (pb+i)->telno);
      }
   }
}
int readin(PhoneBk *pb)
{
   int size=0;
   char *p;

   while (1) {
      printf("Enter name: \n");
      fgets(pb->name, 80, stdin);
      if (p=strchr(pb->name,'\n')) *p = '\0';
      if (strcmp(pb->name,"#")==0)
         break;
      printf("Enter tel: \n");
      fgets(pb->telno, 80, stdin);
      if (p=strchr(pb->telno,'\n')) *p = '\0';
      pb++;
      size++;
   }
   return size;
}
void search(PhoneBk *pb, int size, char *target)
{
   int i;

   for (i=0;i<size;i++,pb++){
      if (strcmp(pb->name,target)==0){
         printf("Name = %s, Tel = %s\n",target,pb->telno);
         break;
      }
   }
   if (i==size)
      printf("Name not found!\n");
}
```