Practice Lab Assignment 7

For this Practice Lab Assignment, you will write programs in C, **making use of the concepts that have been taught in the class.**

Instructions

- There are 16 questions in this assignment.
- Any discussion with neighbor/or any other student is strictly not allowed.
- Mobile phones are not allowed. If found, disciplinary action may be taken.

Due Date: **This is only a Practice Lab so no submission is required.**

Grading Criteria

No Grading Criteria.

Programming Questions

1. **Run the following program and examine the output:**

```
#include<stdio.h>
void main()
{
        int a = 5, *b, **c = &b;
        b = &a;
        *b = 7;
        printf("%d, %d, %d\n", a, *b, **c);
        printf("%p, %p, %p\n", &a, &b, &c);
        printf("%p, %p\n", b, *c);
        printf("%p\n", c);
}
```

2. **Run the following program and examine the output:**

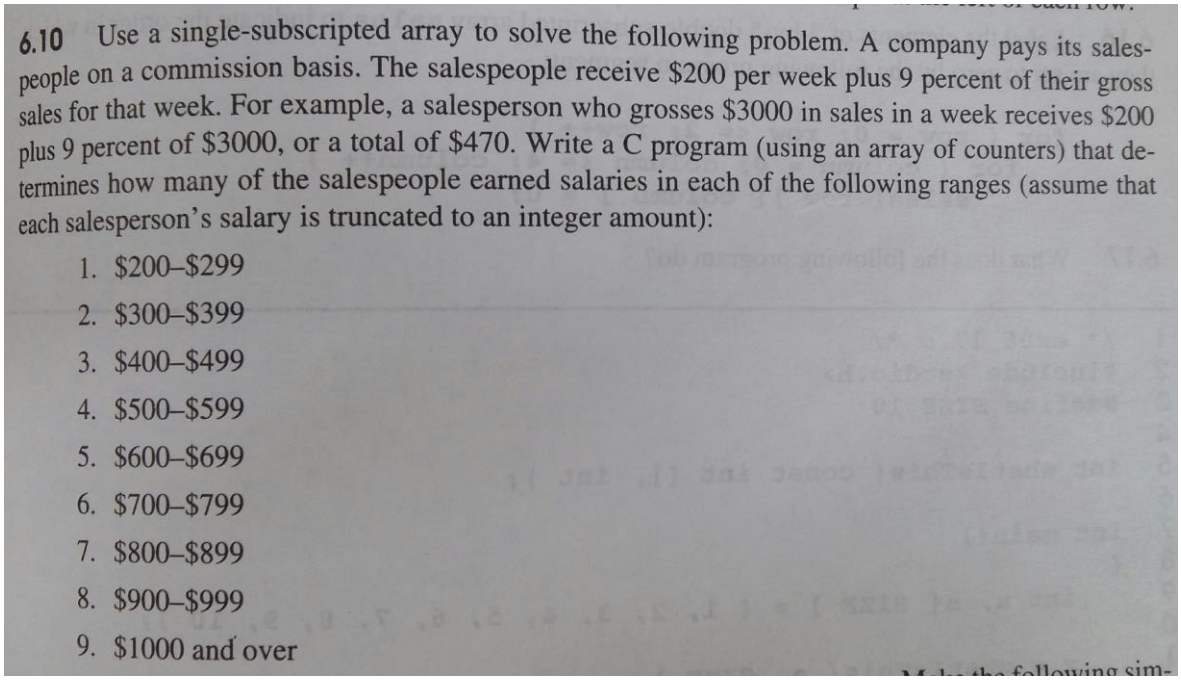
```
#include <stdio.h>
int sum1(int, int);
int sum2(int *, int *);
void sum3(int *, int *, int *);
void main()
{
        int a, b, s;
        a = 5, b = 7;
        printf("Before calling sum1: a = %d, b = %d\n", a, b);
```

```
        s = sum1(a, b);
        printf("After calling sum1: a = %d, b = %d, s = %d\n", a, b, s);
        a = 8, b = 9;
        printf("Before calling sum2: a = %d, b = %d\n", a, b);
        s = sum2(&a, &b);
        printf("After calling sum2: a = %d, b = %d, s = %d\n", a, b, s);
        a = 3, b = 4;
        printf("Before calling sum3: a = %d, b = %d\n", a, b);
        sum3(&a, &b, &s);
        printf("After calling sum3: a = %d, b = %d, s = %d\n", a, b, s);
}

int sum1(int a, int b)
{
        int sum, t;
        sum = a + b;
        t = a;
        a = b;
        b = t;
        return sum;
}

int sum2(int *a, int *b)
{
        int sum, t;
        sum = *a + *b;
        t = *a;
        *a = *b;
        *b = t;
        return sum;
}

void sum3(int *a, int *b, int *s)
{
        int t;
        *s = *a + *b;
        t = *a;
        *a = *b;
        *b = t;
}
```

3. Twenty-Five numbers are entered from the keyboard into an array. Write a program to find out how many of them are positive, how many are negative, how many are even and how many are odd.
4. Implement the following procedure to generate prime numbers from 1 to 100 into a program. This procedure is called sieve of Eratosthenes.
   Step 1 Fill an array **num[100]** with numbers from 1 to 100.
   Step 2 Starting with the second entry in the array, set all its multiples to zero.

Step 3 Proceed to the next non-zero element and set all its multiples to zero.
Step 4 Repeat step 3 till you have set up the multiples of all the non-zero elements to zero.
Step 5 At the conclusion of step 4, all the non-zero entries left in the array would be prime numbers, so print out these numbers.

5. Write a program to copy the contents of one array into another in the reverse order.
6. Write a program to insert a number in an array that is already sorted in ascending order. You need to take the size of the array one extra than the original size.
7. Read in 20 numbers each of which is between 10 and 100, inclusive. As each number is read, store it only if it is not a duplicate of a number already read. Use the smallest possible array to solve this problem.
8. Write a C Program to Put Even & Odd Elements of an Array in 2 Separate Arrays.
9. Write a C Program to Cyclically Permute the Elements of an Array.
10.



6.10 Use a single-subscripted array to solve the following problem. A company pays its salespeople on a commission basis. The salespeople receive $200 per week plus 9 percent of their gross sales for that week. For example, a salesperson who grosses $3000 in sales in a week receives $200 plus 9 percent of $3000, or a total of $470. Write a C program (using an array of counters) that determines how many of the salespeople earned salaries in each of the following ranges (assume that each salesperson's salary is truncated to an integer amount):

1. $200–$299
2. $300–$399
3. $400–$499
4. $500–$599
5. $600–$699
6. $700–$799
7. $800–$899
8. $900–$999
9. $1000 and over

11. Write a program which performs the following tasks:
    - Initialize an integer array of 10 elements in **main()**
    - Pass the entire array to a function **modify()**
    - In **modify()** multiply each element of array by 3
    - Return the control to **main()** and print the new array elements in **main()**

12. Write a program that interchanges the odd and even indexed components of an array.

For Example: If input array is: 3, 5, 1, 8, 2, 9
The output array should be: 5, 3, 8, 1, 9, 2

13. Twenty-Five numbers are entered from the keyboard into an array. The number to be searched is entered through the keyboard by the user. Write a program to find if the number to be searched is present in the array and if it is present, display the number of times it appears in the array.

14. Write a Program to delete all duplicate elements from an array. If a number is occurring more than 1 time, then all the duplicate entries must be deleted and the number must be present only 1 time in the array. Then display the resultant array.

15. Write a program to find whether the array of integers contains a duplicate number. If yes, then print the position of original number and all the duplicate numbers.

16. Write a program to find out a pair of two numbers in an array whose sum is closest to 0.