1. Study Sixth edition Chapter 6, Section 6.2.

2. Translate the following C program to Pep/10 assembly language.

```
int x;
int y;
int main() {
    scanf("%d %d", &x, &y);
    printf("x == %d, y == %d\n", x, y);
    x = x+y;
    y = x-y;
    x = x-y;
    printf("x == %d, y == %d\n", x, y);
    return 0;
}
```

If the input is

```
15 23
```

the output should be

```
x == 15, y == 23
x == 23, y == 15
```

Your assembly language program must contain (1) a documentation section with your name, date, and assignment number at the top of the program, and (2) trace tags for all the variables.

Name your file *xx*prob0614a.pep (all lowercase) where *xx* is your assigned two-digit number. For example, if your two-digit number is 99 you would name it 99prob0614a.pep. Note that the app will automatically append the file extension .pep.

Hand in your file as an attachment in Canvas under Assignment 14.

3. Translate the following C program to Pep/10 assembly language.

```
int main() {
    int x;
    int y;
    scanf("%d %d", &x, &y);
    printf("x == %d, y == %d\n", x, y);
    x = x+y;
    y = x-y;
    x = x-y;
    printf("x == %d, y == %d\n", x, y);
    return 0;
}
```

Your assembly language program must contain (1) a documentation section with your name, date, and assignment number at the top of the program, and (2) trace tags for all the variables.

Name your file *xx*prob0614b.pep (all lowercase) where *xx* is your assigned two-digit number. For example, if your two-digit number is 99 you would name it 99prob0614b.pep. Note that the app will automatically append the file extension .pep.

Hand in your file as an attachment in Canvas under Assignment 14.