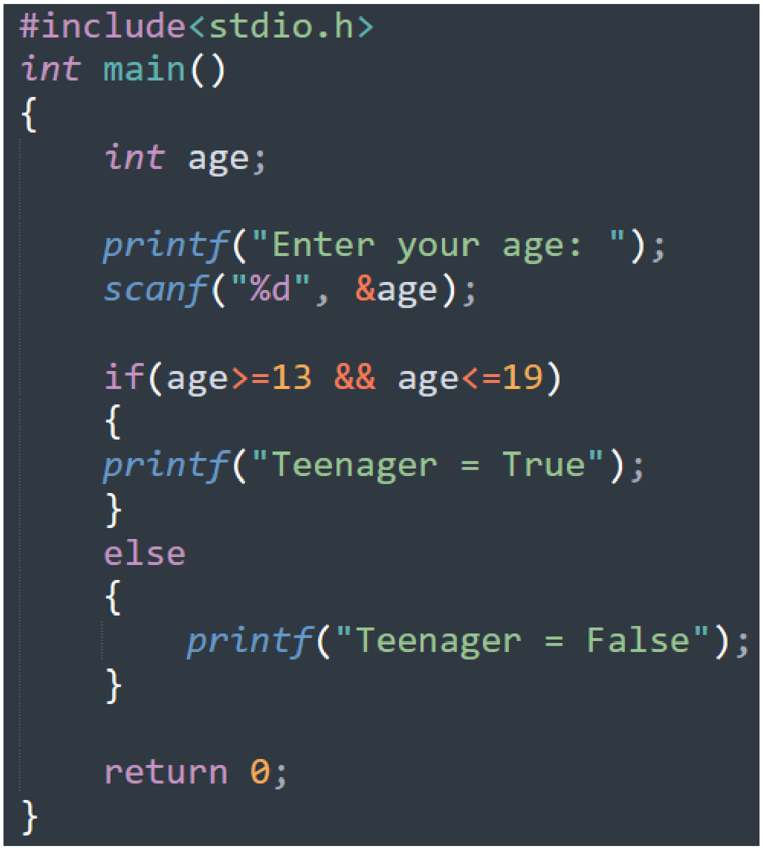
Selection Statements
Lecture 3 Assignments

1. The following `if` statement is unnecessarily complicated. Simplify it as much as possible. (*Hint:* The entire statement can be replaced by a single assignment.)

```
if (age >= 13)
   if (age <= 19)
      teenager = true;
   else
      teenager = false;
else if (age < 13)
   teenager = false;
```

Save your code as `as1.c`

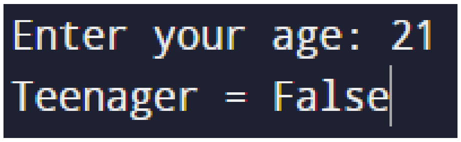


```
#include<stdio.h>
int main()
{
    int age;

    printf("Enter your age: ");
    scanf("%d", &age);

    if(age>=13 && age<=19)
    {
    printf("Teenager = True");
    }
    else
    {
        printf("Teenager = False");
    }

    return 0;
}
```



```
Enter your age: 21
Teenager = False
```

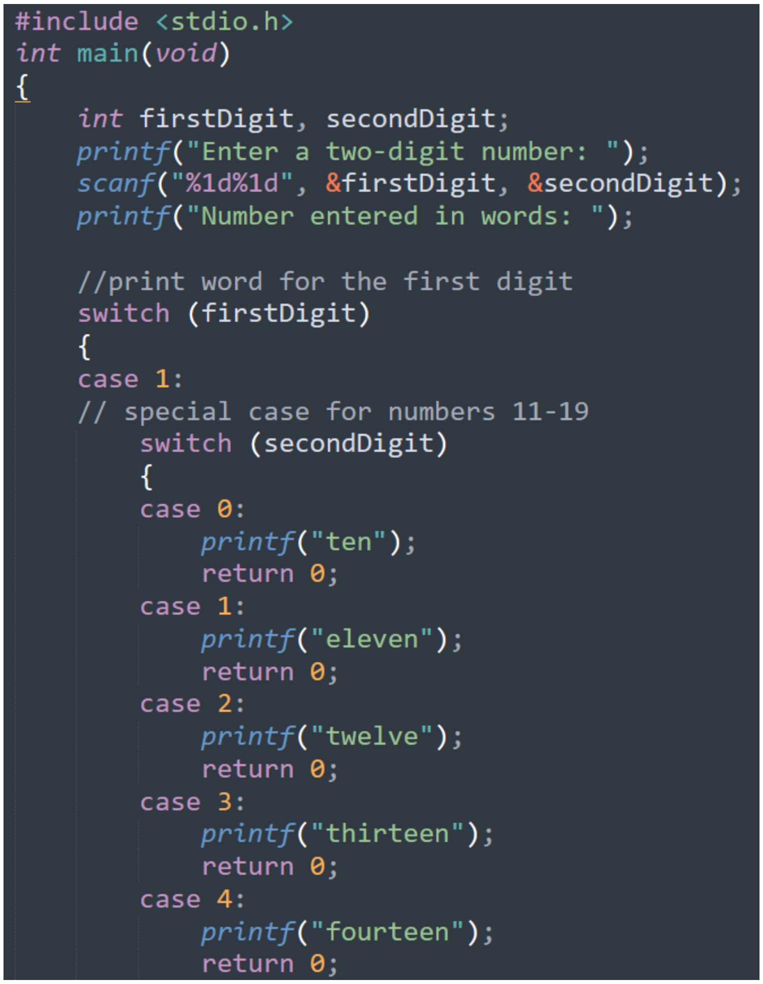
2 Write a C program that does the following:

Enter a two-digit number: 25
Number entered in words: twenty-five

Hint:

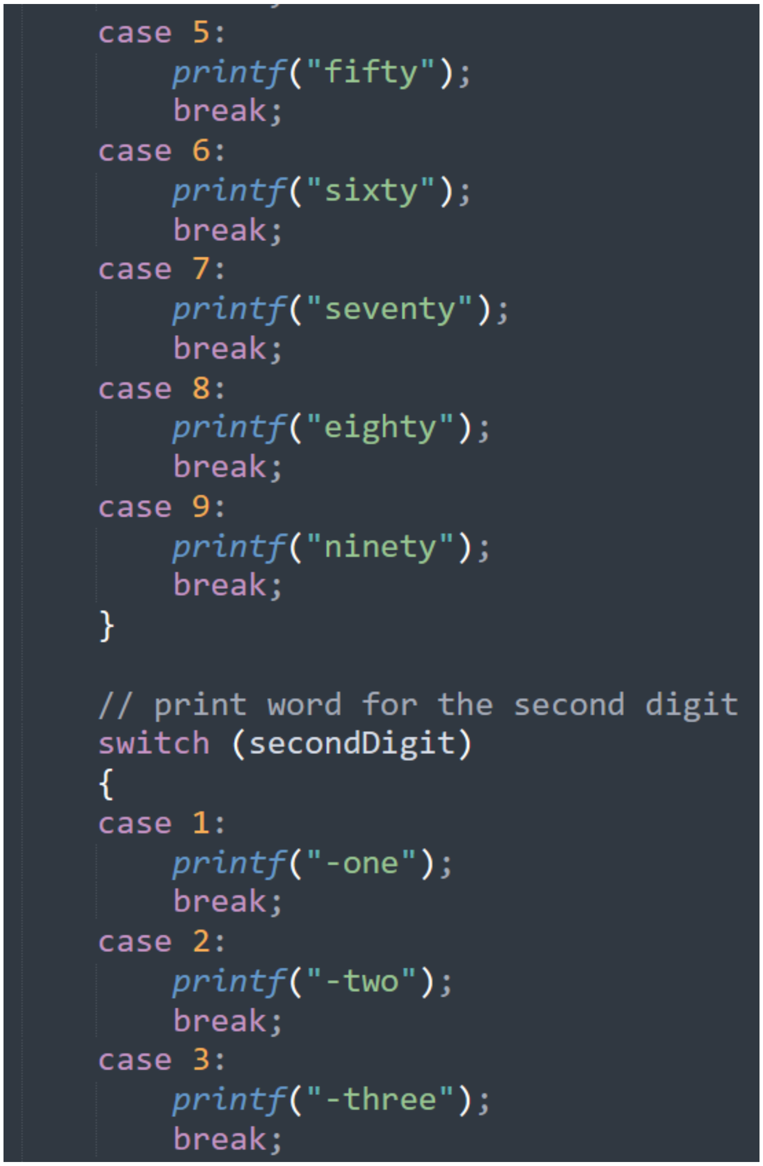
- Break the number into two digits.
- Note: `11` and `19` require special treatment.

Save your code as `as2.c`



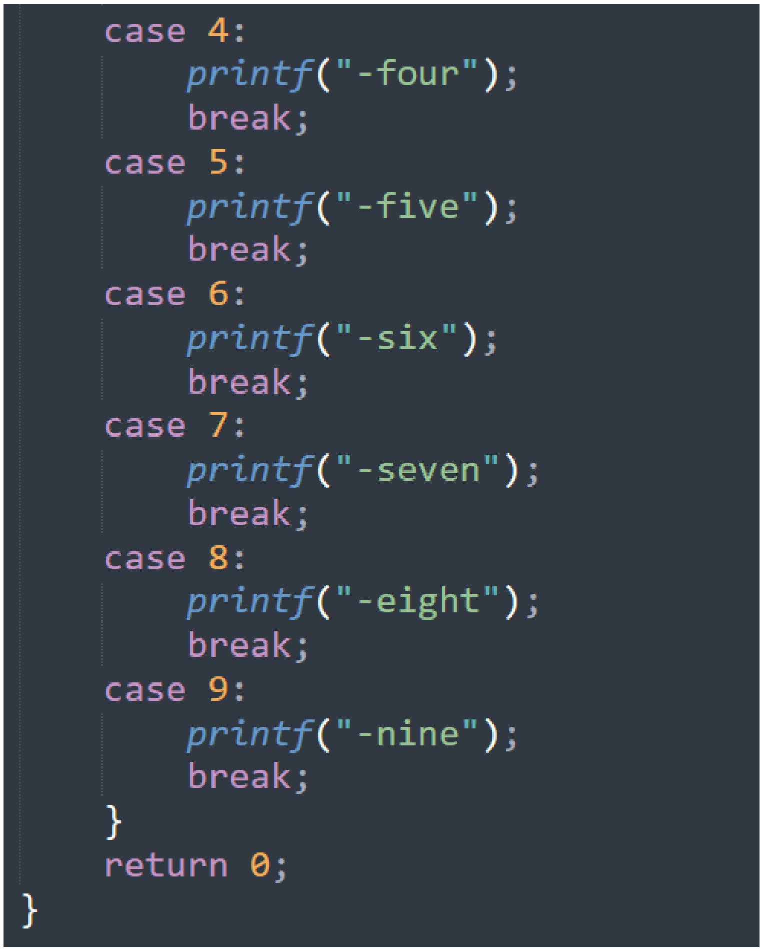
```
#include <stdio.h>
int main(void)
{
    int firstDigit, secondDigit;
    printf("Enter a two-digit number: ");
    scanf("%1d%1d", &firstDigit, &secondDigit);
    printf("Number entered in words: ");

    //print word for the first digit
    switch (firstDigit)
    {
    case 1:
    // special case for numbers 11-19
        switch (secondDigit)
        {
        case 0:
            printf("ten");
            return 0;
        case 1:
            printf("eleven");
            return 0;
        case 2:
            printf("twelve");
            return 0;
        case 3:
            printf("thirteen");
            return 0;
        case 4:
            printf("fourteen");
            return 0;
```

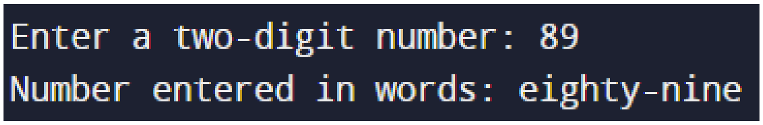


```
    case 5:
        printf("fifty");
        break;
    case 6:
        printf("sixty");
        break;
    case 7:
        printf("seventy");
        break;
    case 8:
        printf("eighty");
        break;
    case 9:
        printf("ninety");
        break;
    }

    // print word for the second digit
    switch (secondDigit)
    {
    case 1:
        printf("-one");
        break;
    case 2:
        printf("-two");
        break;
    case 3:
        printf("-three");
        break;
```



```c
    case 4:
        printf("-four");
        break;
    case 5:
        printf("-five");
        break;
    case 6:
        printf("-six");
        break;
    case 7:
        printf("-seven");
        break;
    case 8:
        printf("-eight");
        break;
    case 9:
        printf("-nine");
        break;
    }
    return 0;
}
```



```
Enter a two-digit number: 89
Number entered in words: eighty-nine
```

Instructions for submissions

- Take screenshots of your codes for numbers which requires coding (e.g., 1, 2, 3) and embed it on the pdf along with an example output.
- Submit your answers in a pdf file with filename assignment2[surname].pdf
- Save the pdf file (assignment3[surname].pdf) and the codes in the directory:
  CMSC21/Lecture3/Assignments/
- Remember that you have initially created this repository for your reading assignment.
- Upload to github.
  - Download git cmd
  - Navigate to the CMSC21 Folder
  - For example (assuming your CMSC21 folder is in Documents)
    - `cd Documents/CMSC21`
    - `git add –all`
    - `git commit -m “Lecture 3 Assignment”`
    - `git push -u origin main`