

Selection Statements

Lecture 3 Assignments

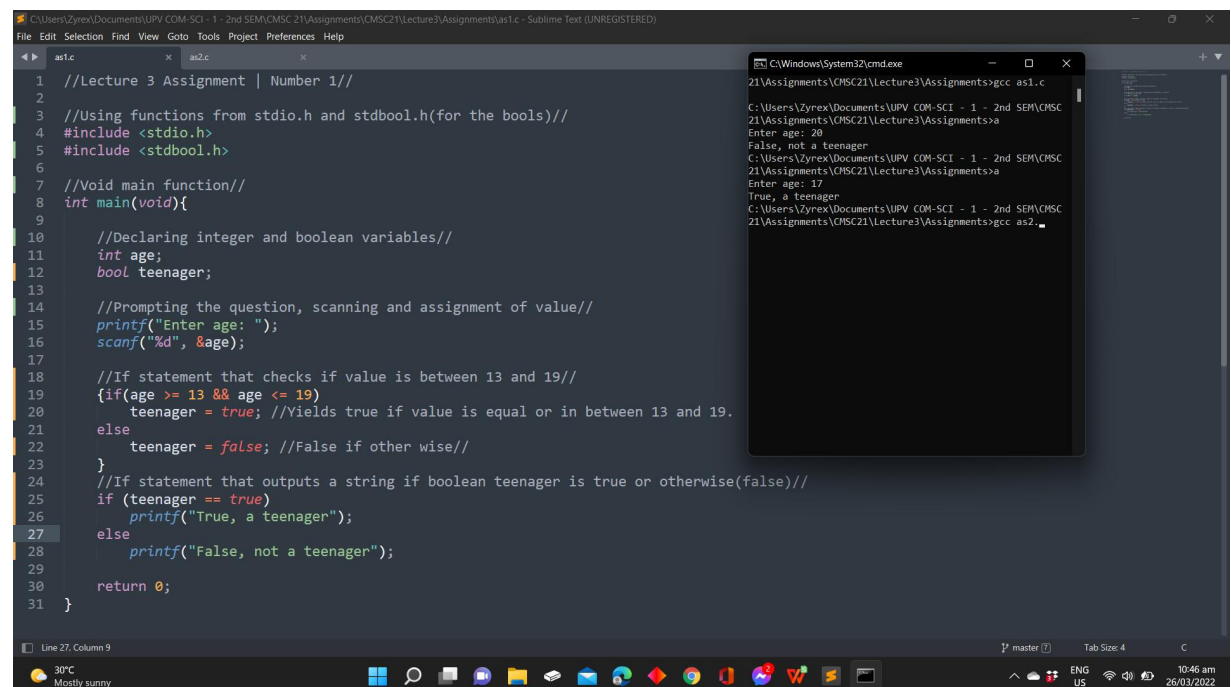
By Zyrex Djewel Ganit

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



The screenshot shows a Sublime Text editor window with a file named 'as1.c'. The code in the editor is as follows:

```
1 //Lecture 3 Assignment | Number 1//
2
3 //Using functions from stdio.h and stdbool.h(for the bools)//
4 #include <stdio.h>
5 #include <stdbool.h>
6
7 //Void main function//
8 int main(void){
9
10     //Declaring integer and boolean variables//
11     int age;
12     bool teenager;
13
14     //Prompting the question, scanning and assignment of value//
15     printf("Enter age: ");
16     scanf("%d", &age);
17
18     //If statement that checks if value is between 13 and 19//
19     {if(age >= 13 && age <= 19)
20         teenager = true; //Yields true if value is equal or in between 13 and 19.
21     else
22         teenager = false; //False if other wise//
23     }
24     //If statement that outputs a string if boolean teenager is true or otherwise(false)//
25     if (teenager == true)
26         printf("True, a teenager");
27     else
28         printf("False, not a teenager");
29
30     return 0;
31 }
```

Overlaid on the right side of the editor is a Windows Command Prompt window. It shows the execution of the program:

```
C:\Users\Zyrex\Documents\UPV COM-SCI - 1 - 2nd SEM\CMSC 21\Assignments\CMSC21\Lecture3\Assignments>gcc as1.c
C:\Users\Zyrex\Documents\UPV COM-SCI - 1 - 2nd SEM\CMSC 21\Assignments\CMSC21\Lecture3\Assignments>a
Enter age: 20
False, not a teenager
C:\Users\Zyrex\Documents\UPV COM-SCI - 1 - 2nd SEM\CMSC 21\Assignments\CMSC21\Lecture3\Assignments>a
Enter age: 17
True, a teenager
C:\Users\Zyrex\Documents\UPV COM-SCI - 1 - 2nd SEM\CMSC 21\Assignments\CMSC21\Lecture3\Assignments>gcc as2.
```

The Windows taskbar at the bottom shows the system clock as 10:48 am on 26/03/2022, with a temperature of 30°C and 'Mostly sunny' weather.

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

The screenshot shows a Windows desktop with a Sublime Text editor and a Windows Command Prompt. The Sublime Text editor is open to a file named `as2.c`, which contains a C program that takes a two-digit number as input and prints its English representation. The program uses nested switch statements to handle the tens and units digits, with special cases for numbers like 11 and 19. The Windows Command Prompt is running the program, showing the input '25' and the output 'twenty-five'.

```
#include <stdio.h>

//Main function//
int main(void){
    //Declaring the variables//
    int digit1, digit2;

    //Prompting question and assigning values to variable//
    printf("Enter a two-digit number: ");
    scanf("%i%i", &digit1, &digit2);

    //Body of if statement with nested switch will be executed if the first
    digit equals to one as it need special treatment//
    if (digit1 == 1) {
        switch(digit2 % 10){ //2nd digit must be base 10 numeral as a condition//
            case 0: printf("ten"); break;
            case 1: printf("eleven"); break;
            case 2: printf("twelve"); break;
            case 3: printf("thirteen"); break;
            case 4: printf("fourteen"); break;
            case 5: printf("fifteen"); break;
            case 6: printf("sixteen"); break;
            case 7: printf("seventeen"); break;
            case 8: printf("eighteen"); break;
            case 9: printf("nineteen"); break;
        }
        return 0;
    }

    //Switch case for the 1st digit//
    switch(digit1 % 10) {
        case 1: printf("ten"); break;
        case 2: printf("twenty"); break;
        case 3: printf("thirty"); break;
        case 4: printf("forty"); break;
        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;
    }

    //Switch case for the second digit//
    switch(digit2 % 10) {
        case 0: break; //if zero, it will not print anything (we cannot say twenty-zero)//
        case 1: printf("-one"); break;
        case 2: printf("-two"); break;
        case 3: printf("-three"); break;
        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;
    }
}
```

Windows Command Prompt Output:

```
C:\Users\Zyrex\Documents\UPV COM-SCI - 1 - 2nd SEM\CMSC 21\Assignments>gcc as1.c
C:\Users\Zyrex\Documents\UPV COM-SCI - 1 - 2nd SEM\CMSC 21\Assignments>a
Enter age: 20
False, not a teenager
C:\Users\Zyrex\Documents\UPV COM-SCI - 1 - 2nd SEM\CMSC 21\Assignments>a
Enter age: 17
True, a teenager
C:\Users\Zyrex\Documents\UPV COM-SCI - 1 - 2nd SEM\CMSC 21\Assignments>gcc as2.c
C:\Users\Zyrex\Documents\UPV COM-SCI - 1 - 2nd SEM\CMSC 21\Assignments>a
Enter a two-digit number: 25
twenty-five
C:\Users\Zyrex\Documents\UPV COM-SCI - 1 - 2nd SEM\CMSC 21\Assignments>
```