

C Programming Language

IF - Else

Dr. Mostafa A. Elhosseini

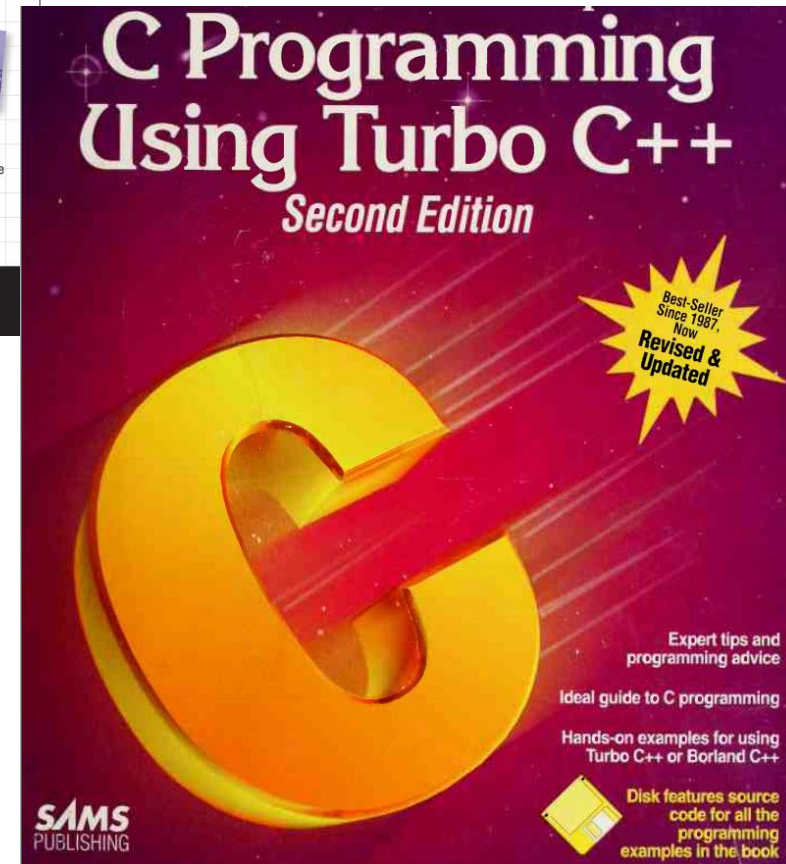
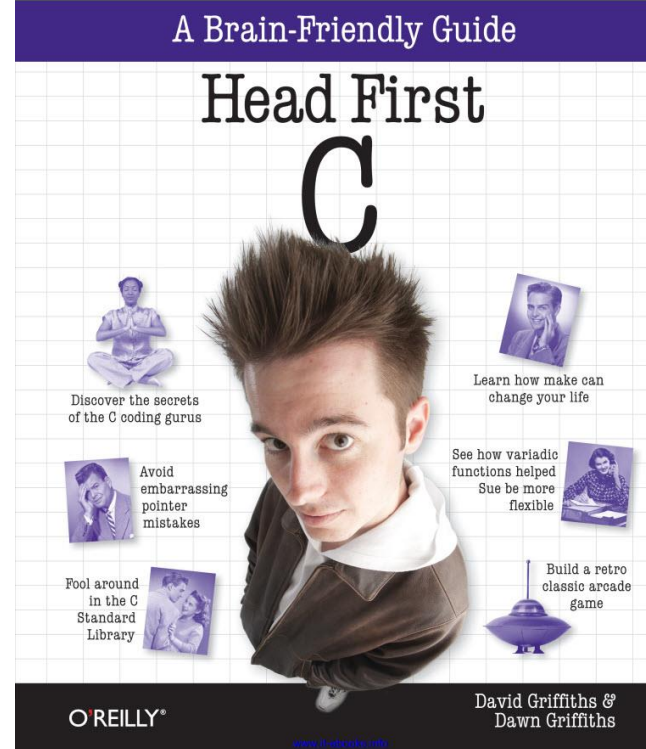
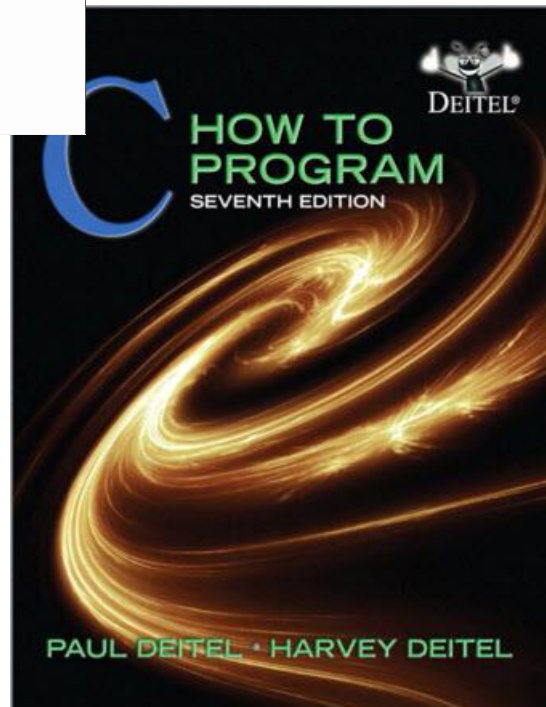
SECOND EDITION

THE



PROGRAMMING LANGUAGE

BRIAN W. KERNIGHAN
DENNIS M. RITCHIE



Revise

✗ Pseudocode

✗ Flowchart

✗ Relational operator

✗ If statement

- Surround condition with **parenthesis**
- No **then** keyword after condition
- **No semicolon** after if statement
- If only one statement to be executed, just write the statement
 - If many, use the begin and end braces

Agenda

- ✂ Conditional statement
 - if-else statement

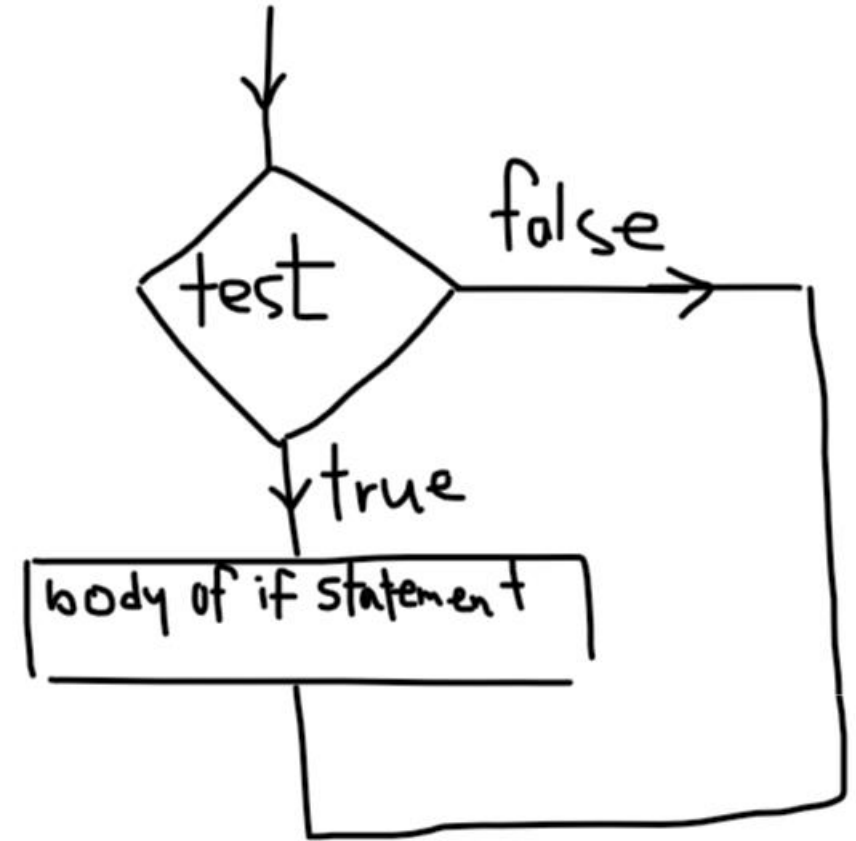
if statement

Keyword
if (x == 10)
{
printf("perfect");
}

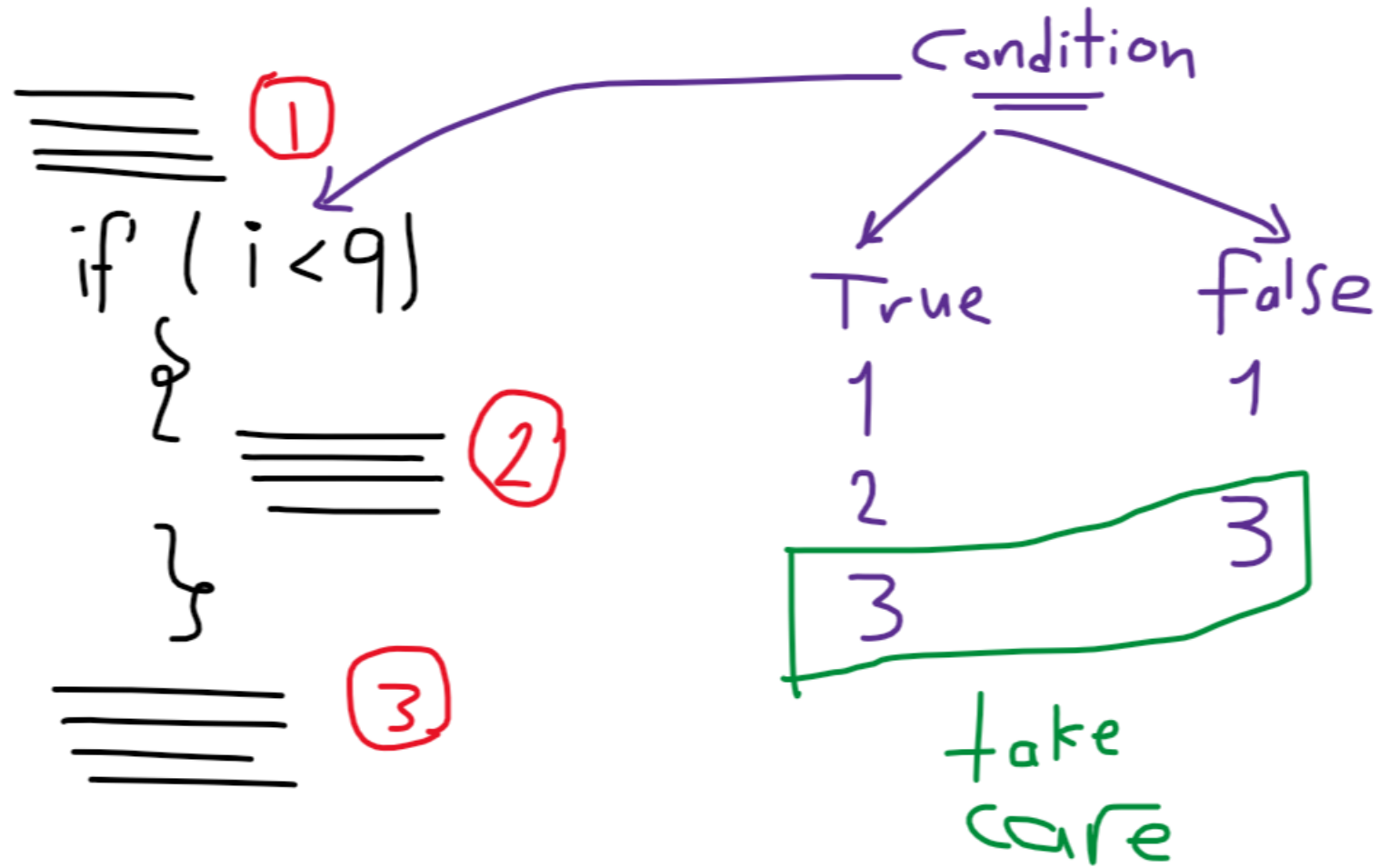
Relational expression
True
false

body

no semicolon here

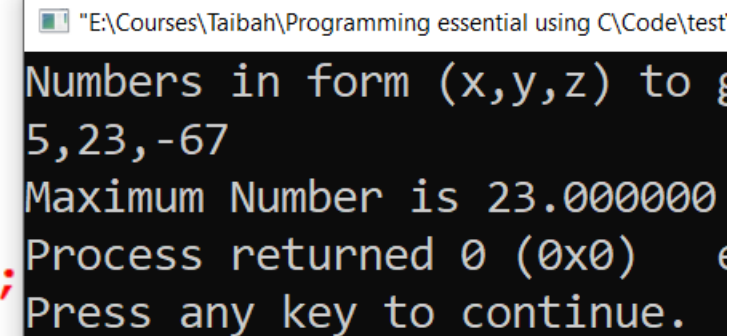


■ if-statement



If-statement

```
1  #include <stdio.h>
2  int main()
3  {
4      float x,y,z,mymax;
5      puts("Numbers in form (x,y,z) to get maximum:");
6      scanf("%f,%f,%f",&x,&y,&z);
7      mymax = x;
8      if(mymax<y) mymax = y;
9      if(mymax<z) mymax = z;
10     printf("Maximum Number is %f",mymax);
11     return 0;
12 }
```



E:\Courses\Taibah\Programming essential using C\Code\test

Numbers in form (x,y,z) to get maximum:
5,23,-67
Maximum Number is 23.000000
Process returned 0 (0x0)
Press any key to continue.

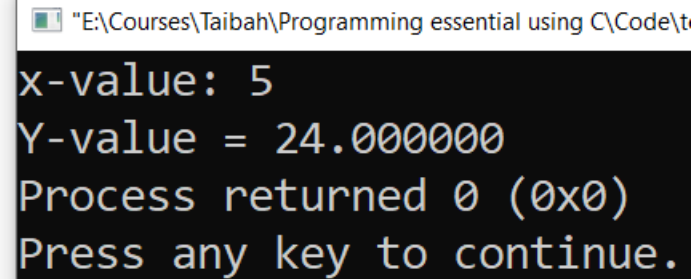
If statement

```
1  #include <stdio.h>
2  #include <math.h>
3  main()
4  {
5      float x, y;
6      printf("x-value: ");
7      scanf("%f", &x);
8      if (x <= 4)
9          y = pow(x, 2) + 7;
10     if (x > 4)
11         y = x + 19;
12     printf("Y-value = %f", y);
13 }
```

$$y(x) = \begin{cases} x^2 + 7 & x \leq 4 \\ x + 19 & \text{otherwise} \end{cases}$$

If-statement

```
1  #include <stdio.h>
2  #include <math.h>
3  main()
4  {
5      float x,y;
6      printf("x-value: ");
7      scanf("%f",&x);
8      y = x+19;
9      if(x<=4)
10         y = pow(x,2)+7;
11     printf("Y-value = %f",y);
12 }
```



E:\Courses\Taibah\Programming essential using C\Code\t
x-value: 5
Y-value = 24.000000
Process returned 0 (0x0)
Press any key to continue.

■ Some on characters

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 int main()
4 {
5     char ch,n;
6     puts("Character in small case: ");
7     ch = getchar();
8     n = toascii(ch);
9     printf("\n you typed %c\t Ascii: %x",ch,n);
10    printf("\n upper %c\t Ascii: %x",toupper(ch),toascii(toupper(ch)))
11 }
12
```

"E:\Courses\Taibah\Programming essential using C\Code\test\bin\Debug\test.exe"

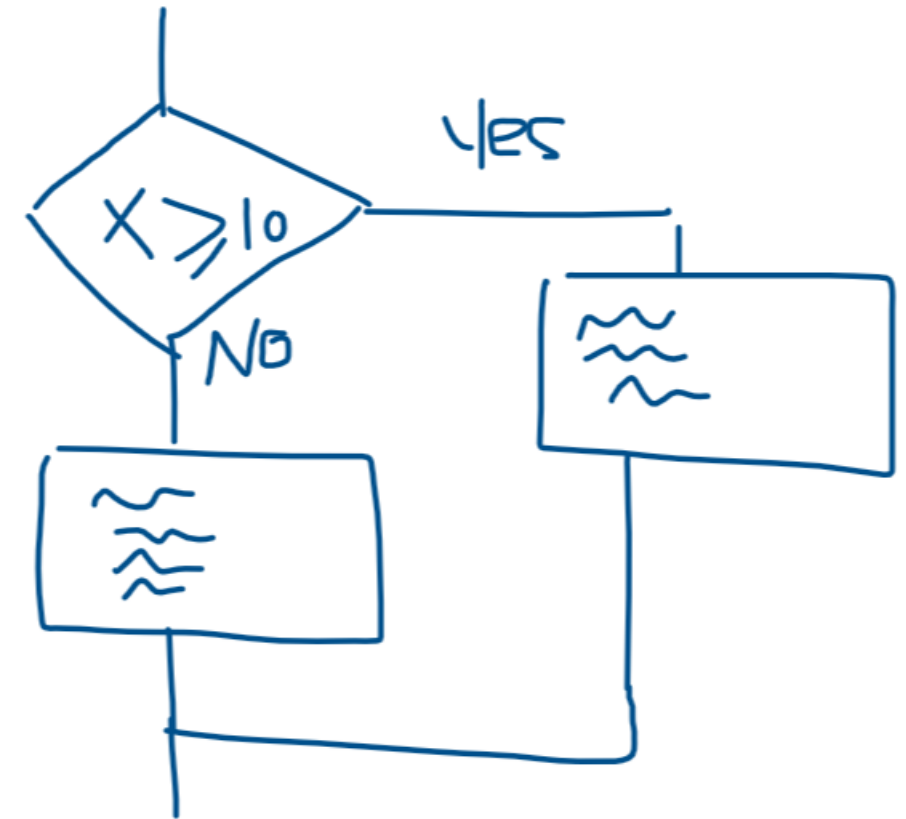
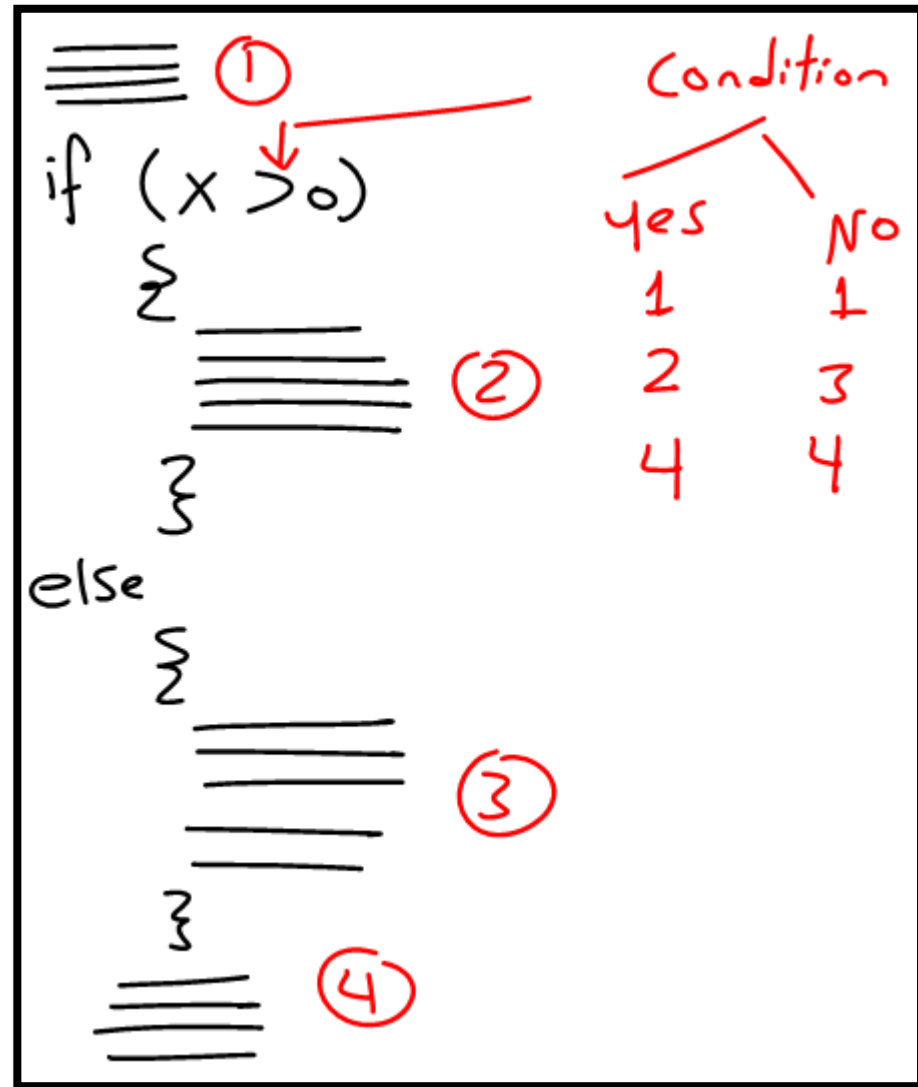
Character in small case:

a

you typed a Ascii: 61

upper A Ascii: 41

If – else statement



Example: Pass // Fail

```
1  #include <stdio.h>
2  #include <math.h>
3  main()
4  {
5      float x;
6      printf("score: ");
7      scanf("%f", &x);
8      if(x<60)
9          printf("Fail");
10     else
11         printf("Pass");
12 }
```

Example

- Even // odd

```
1  #include <stdio.h>
2  main()
3  {
4      int x;
5      puts("Number: ");
6      scanf("%d", &x);
7      if(x%2 == 0)
8          printf("Even");
9      else
10         printf("Odd");
11 }
```

```
1  #include <stdio.h>
2  main()
3  {
4      int x;
5      puts("Number: ");
6      scanf("%d", &x);
7      if(x%2 == 0) printf("Even");
8      else printf("Odd");
9  }
```

Two branch function

```
2  #include <math.h>
3  main()
4  {
5      float x, y;
6      printf("x-value: ");
7      scanf("%f", &x);
8      if(x<=4)
9          y = pow(x, 2)+7;
10     else
11         y = x+19;
12     printf("Y-value = %f", y);
13 }
```

$$y(x) = \begin{cases} x^2 + 7 & x \leq 4 \\ x + 19 & \text{otherwise} \end{cases}$$

Quadratic equation

$$ax^2 + bx + c = 0$$
$$x_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

```
1  #include <stdio.h>
2  #include <math.h>
3  int main()
4  {
5      float a,b,c,D,x1,x2;
6      puts("quadratic coefficients: (a,b,c)");
7      scanf("%f,%f,%f",&a,&b,&c);
8      D = pow(b,2)-4*a*c;
9      if(D>=0)
10     {
11         x1 = (-b+sqrt(D))/(2*a);
12         x2 = (-b-sqrt(D))/(2*a);
13         printf("x1 = %f\t x2 = %f ",x1,x2);
14     }
15     else
16         printf("Imaginary roots");
17 }
```

Logical operator

- `&&` → AND
- `||` → OR
- `!` → Not

```
1  #include <stdio.h>
2  #include <math.h>
3  main()
4  {
5      float x;
6      puts("x-number: ");
7      scanf("%f", &x);
8      if(x>=30 && x<=40)
9          puts("In range");
10     else
11         puts("Out of range");
12 }
```

What if gets the else?

```
2  #include <math.h>
3  main()
4  {
5      int temp;
6      printf("please type in the temperature: ");
7      scanf("%d",&temp);
8      if(temp < 28)
9          if(temp > 21)
10             printf("Nice Day");
11      else
12         printf("Sure is hot");
13 }
```

What if gets the else?

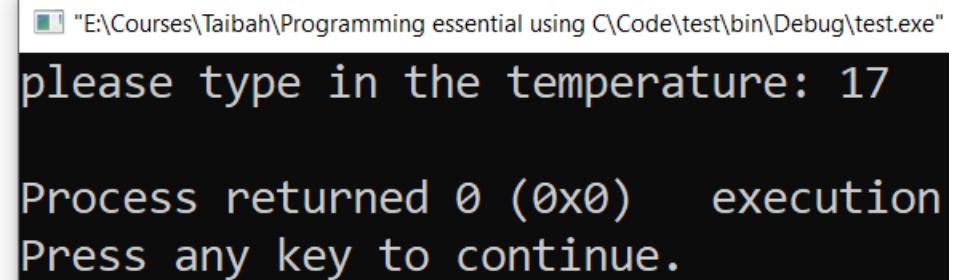
```
2  #include <math.h>
3  main()
4  {
5      int temp;
6      printf("please type in the temperature: ");
7      scanf("%d",&temp);
8      if(temp < 28)
9          if(temp > 21)
10             printf("Nice Day");
11     else
12         printf("Sure is hot");
13 }
```

"E:\Courses\Taibah\Programming essential using C\Code\test\bin\Debug\test.exe"

```
please type in the temperature: 17
Sure is hot
Process returned 0 (0x0)   execution
Press any key to continue.
```

What if gets the else?

```
1  #include <stdio.h>
2  #include <math.h>
3  main()
4  {
5      int temp;
6      printf("please type in the temperature: ");
7      scanf("%d", &temp);
8      if(temp < 28) {
9          if(temp > 21)
10             printf("Nice Day");
11     }
12     else
13         printf("Sure is hot");
14 }
```



"E:\Courses\Taibah\Programming essential using C\Code\test\bin\Debug\test.exe"

please type in the temperature: 17

Process returned 0 (0x0) execution
Press any key to continue.