



Programming Fundamentals

With Python



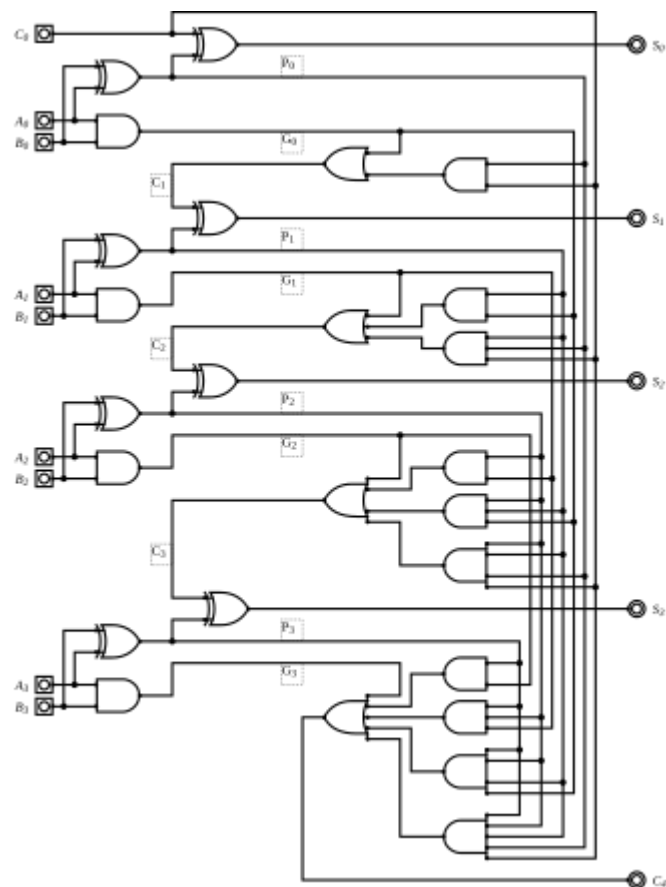
Chapter 3





01

Conditional Logic





Basic Logic Operators

NOT

x	F
0	1
1	0



AND

x	y	F
0	0	0
0	1	0
1	0	0
1	1	1



OR

x	y	F
0	0	0
0	1	1
1	0	1
1	1	1





English vs. Boolean Logic

I will run outside if it is not raining and it is the weekend.

X = raining

Y = weekend

The boolean logic is Result = **Not X and Y**

Raining	Weekend	Result
True	True	False
True	False	False
False	True	True
False	False	False



English vs. Boolean Logic

Write a logic expression to check if x and y are less than 5 but greater than or equal to 3:

Did you get $3 \leq x \ \&\& \ y < 5$? WRONG

Did you get $(3 \leq x < 5) \ \&\& \ (3 \leq y < 5)$? WRONG

Answer:

$3 \leq x \ \&\& \ x < 5 \ \&\& \ 3 \leq y \ \&\& \ y < 5$



English vs. Boolean Logic

Write a logic expression to check if x is even and y is odd:

Answer:

```
x % 2 == 0 && y % 2 != 0
```

NOTE: The modulus operator divides then outputs the remainder. So, if $x = 5$ then

$x \% 2 = 1$ because $5 / 2 = 2$ plus remainder = 1, which is odd since it is not equal to 0.



Order of Operations

1. Function calls and brackets
2. Not Operator
3. Arithmetic
 - a. $*$, $/$, $\%$
 - b. $+$, $-$
4. Comparison: $<$, $<=$, $>=$, $>$, $==$, $!=$
5. Logical: OR, AND
6. Assignment: $=$

NOTE: It is easy to forget that $=$ is for comparison and $=$ is for assignment

NOTE: Once logical operators are used in the expression, the output is always boolean (0 for false and 1 for True)



Order of Operations

```
1  x = 1
2  y = 0
3
4  r= not(x - 1 + y * 1 and not y - 1) and (x + 3 > 1 or y + 1)
5
6  print(r)
```

Output

True



Logical Expressions Example

1 $x = 1$

2 $y = 0$

```
r = not(x - 1 + y * 1 and not y - 1) and (x + 3 > 1 or y + 1)
```

```
r = not(x - 1 + y * 1 and 1 - 1) and (x + 3 > 1 or y + 1)
```

```
r = not(x - 1 + 0 and 1 - 1) and (x + 3 > 1 or y + 1)
```

```
r = not(0 + 0 and 0) and (4 > 1 or 1)
```

```
r = not(0) and (1 or 1)
```

```
r = not(0) and (1 or 1)
```

```
r = 1 and 1
```

```
r = 1
```



02

If Statement



```
if ($_POST['user_password_new'] === $_POST['user_password_repeat']) {  
    if (strlen($_POST['user_password_new']) > 5) {  
        if (strlen($_POST['user_name']) < 65 && strlen($_POST['user_name']) > 1) {  
            if (preg_match('/^[a-z\d]{2,64}$/i', $_POST['user_name'])) {  
                $user = read_user($_POST['user_name']);  
                if (!isset($user['user_name'])) {  
                    if ($_POST['user_email']) {  
                        if (strlen($_POST['user_email']) < 65) {  
                            if (filter_var($_POST['user_email'], FILTER_VALIDATE_EMAIL)) {  
                                create_user();  
                                $_SESSION['msg'] = 'You are now registered so please login';  
                                header('Location: ' . $_SERVER['PHP_SELF']);  
                                exit();  
                            } else $msg = 'You must provide a valid email address';  
                        } else $msg = 'Email must be less than 64 characters';  
                    } else $msg = 'Email cannot be empty';  
                } else $msg = 'Username already exists';  
            } else $msg = 'Username must be only a-z, A-Z, 0-9';  
        } else $msg = 'Username must be between 2 and 64 characters';  
    } else $msg = 'Password must be at least 6 characters';  
}
```



Basic Structure

```
1 x = 2
2
3 if x == 0:
4     print("x is equal to 0")
5 elif x > 1:
6     print("x is greater than 1")
7 else:
8     print("x is less than or equal to 1")
```

This is the boolean expression we were talking about. If the result is true it goes in, otherwise, it skips that inward line.

The structure of the logical branches will always be **if** then **else if** then **else** (must be in order). You can also just do **if** or **if** then **else if**.



Nested If statements

```
1 x = 1
2
3 if (x == 0):
4     if (x == 1):
5         print("x is equal to 1")
6     else:
7         print("x is equal to 0")
8 else:
9     print("x is not 0 and x is not 1")
```

You can have if statements within if statements to represent other cases that can only be covered if the outer case is true.

output: x is not 0 and x is not 1

This was a trick. Be careful, you must consider all possible test cases when dealing with if statements.



If statements Example

```
1 age = 18
2
3 ▾ if age <= 12:
4     print("You are a child")
5 ▾ elif age < 60:
6     print("You are an adult")
7 ▾ elif age <= 19:
8     print("You are a teenager")
9 ▾ else:
10    print("You are a senior")
```

Output: You are an adult

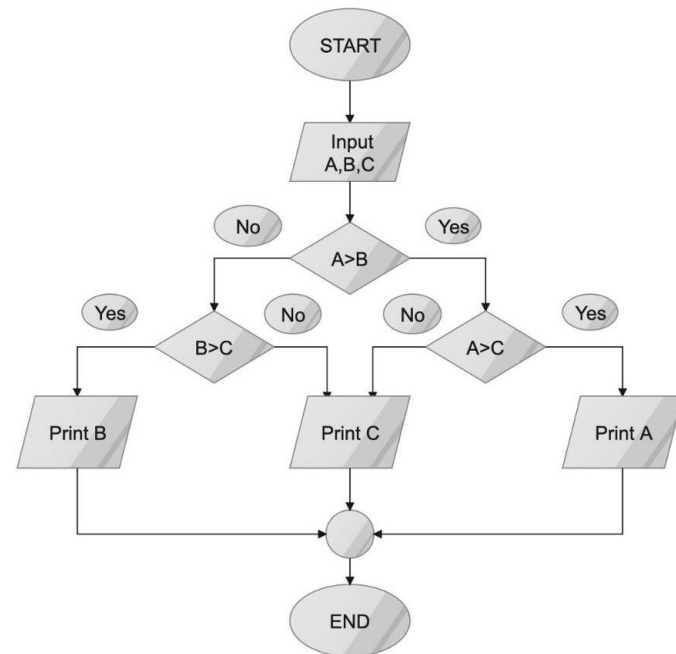
This is another trick. Be careful, order of the if statements matter! Technically, if you are 18, you are less than 60 years old, but you never checked if you were less than or equal to 19 years old.



If statements Tips

When you are creating a program with conditions:

1. Ask what are all the possible cases?
2. Create a flow chart following the logic.
3. Remember that order of the logic matters.
4. Try to reduce the number of if statements and branching as much as possible. The less if statements for the same logic, the better. Sometimes, the logic might seem like it needs nested branching, but you can actually reduce it to an if statement.
5. Make the logic as easy as possible to follow.
6. Omit redundant if statements.





Exercise 1

I will run outside if it is not raining and it is the weekend. Code a program that will tell me if I should run based on the conditions given previously. Make a running and weekend variable and as for user to input values for these variables. The user will input 0 for False and 1 for True.

```
1  raining = int(input("Is it raining today? "))
2  weekend = int(input("Is it the weekend today? "))
3
4  if not raining and weekend:
5      print("You should go running today.")
6  else:
7      print("You should not go running today.")
```



Exercise 2

Write a program that divides two numbers. Make sure it declares numerator, denominator, and result variables. Check to make sure denominator is not 0. Also, assume the program can only handle numbers smaller than or equal to 100.



```
1  numerator = 4
2  denominator = 0
3
4  if (numerator <= 100 and denominator <= 100):
5      if (denominator != 0):
6          result = numerator/denominator
7          print(f"The answer is {result}")
8      else:
9          print("You cannot divide by 0")
10 else:
11     print("The numerator or denominator is too big of a
        number")
```



Exercise 3

Max wants to find his average quiz mark for one of his classes at school. The teacher is nice and will drop the lowest quiz mark out of 3 quizzes. All quizzes were out of 10. Help Max find his average mark.

Possible
Answer

```
1  q1 = float(input("Enter Quiz 1 mark: "))
2  q2 = float(input("Enter Quiz 2 mark: "))
3  q3 = float(input("Enter Quiz 3 mark: "))
4
5  if (q1 <= q2 and q1 <= q3):
6      avgQuizMark = (q2+q3)/2
7  if (q2 <= q1 and q2 <= q3):
8      avgQuizMark = (q1+q3)/2
9  if (q3 <= q1 and q3 <= q2):
10     avgQuizMark = (q1+q2)/2
11
12 print(f"Your average quiz mark is: {avgQuizMark}")
```



Exercise 3

```
1 q1 = float(input("Enter Quiz 1 mark: "))
2 q2 = float(input("Enter Quiz 2 mark: "))
3 q3 = float(input("Enter Quiz 3 mark: "))
4
5 if (q1 <= q2 and q1 <= q3):
6     avgQuizMark = (q2+q3)/2
7 if (q2 <= q1 and q2 <= q3):
8     avgQuizMark = (q1+q3)/2
9 if (q3 <= q1 and q3 <= q2):
10    avgQuizMark = (q1+q2)/2
11
12 print(f"Your average quiz mark is: {avgQuizMark}")
```

But there are some problems with this. There is redundant logic.

Do you really have to assign a certain value to avgQuizMark for n quizzes?

Do you really have to check each quiz with all other quizzes? What if you are calculating class average and there are 200 quizzes to check?

Try to find a better solution! **Hint:** Create a lowest mark variable and keep updating the lowest mark.



Exercise 3

Better Answer

```
1 q1 = float(input("Enter Quiz 1 mark: "))
2 q2 = float(input("Enter Quiz 2 mark: "))
3 q3 = float(input("Enter Quiz 3 mark: "))
4
5 # Assume quiz 1 is the lowest mark
6 lowestQuizMark = q1
7
8 if (q2 < lowestQuizMark):
9     lowestQuizMark = q2
10
11 if (q3 < lowestQuizMark):
12     lowestQuizMark = q3
13
14 avgQuizMark = (q1 + q2 + q3 - lowestQuizMark)/2
15
16 print(f"Your average quiz mark is: {avgQuizMark}")
```



Exercise 4

Now Max wants to sort his quiz marks for lowest to highest.

Possible
Answer

```
1 q1 = float(input("Enter Quiz 1 Mark: "))
2 q2 = float(input("Enter Quiz 2 Mark: "))
3 q3 = float(input("Enter Quiz 3 Mark: "))
4
5 if q1 < q2:
6     if q1 < q3:
7         if q2 < q3:
8             print(q1,q2,q3)
9         else:
10            print(q1,q3,q2)
11    else:
12        print(q3,q1,q2)
13 else:
14     if q1 > q3:
15         if q2 > q3:
16             print(q3,q2,q1)
17         else:
18             print(q2,q3,q1)
19    else:
20        print(q2,q1,q3)
```

```
1 q1 = float(input("Enter Quiz 1 Mark: "))
2 q2 = float(input("Enter Quiz 2 Mark: "))
3 q3 = float(input("Enter Quiz 3 Mark: "))
4
5 if q1 < q2:
6     if q1 < q3:
7         if q2 < q3:
8             print(q1,q2,q3)
9         else:
10            print(q1,q3,q2)
11    else:
12        print(q3,q1,q2)
13 else:
14     if q1 > q3:
15         if q2 > q3:
16             print(q3,q2,q1)
17         else:
18             print(q2,q3,q1)
19    else:
20        print(q2,q1,q3)
```

Again the code has some problems.

There are too many nested if statements, complicating the logic. Also, if we want to sort 1000 quizzes, it would not be feasible to do it this way.

Again, think of a smarter way to do this code.

Hint: What are all the combinations of comparisons we need to make for 3 quizzes? Imagine they reserve a spot and you can swap them.



Better
Answer

```
1  q1 = float(input("Enter Quiz 1 Mark: "))
2  q2 = float(input("Enter Quiz 2 Mark: "))
3  q3 = float(input("Enter Quiz 3 Mark: "))
4
5  if q1 > q2:
6      temp = q1
7      q1 = q2
8      q2 = temp
9
10 if q1 > q3:
11     temp = q1
12     q1 = q3
13     q3 = temp
14
15 if q2 > q3:
16     temp = q2
17     q2 = q3
18     q3 = temp
19
20 print(q1,q2,q3)
```



```
1 q1 = float(input("Enter Quiz 1 Mark: "))
2 q2 = float(input("Enter Quiz 2 Mark: "))
3 q3 = float(input("Enter Quiz 3 Mark: "))
4
5 if q1 > q2:
6     temp = q1
7     q1 = q2
8     q2 = temp
9
10 if q1 > q3:
11     temp = q1
12     q1 = q3
13     q3 = temp
14
15 if q2 > q3:
16     temp = q2
17     q2 = q3
18     q3 = temp
19
20 print(q1,q2,q3)
```

This is what is called the bubble sort algorithm. You check every slot with every other slot except behind it. Since we want to order from smallest to largest, if there exists a value larger than a later value, obviously it needs to be swapped.

NOTE: This swapping technique is common in software engineering and computer science algorithms.


```

1  q1 = float(input("Enter Quiz 1 Mark: "))
2  q2 = float(input("Enter Quiz 2 Mark: "))
3  q3 = float(input("Enter Quiz 3 Mark: "))
4
5  if q1 > q2:
6      temp = q1
7      q1 = q2
8      q2 = temp
9
10 if q1 > q3:
11     temp = q1
12     q1 = q3
13     q3 = temp
14
15 if q2 > q3:
16     temp = q2
17     q2 = q3
18     q3 = temp
19
20 print(q1,q2,q3)

```

?

Example

9.5	6	8
-----	---	---

temp	9.5
------	-----

6	9.5	8
---	-----	---

temp	9.5
------	-----

6	8	9.5
---	---	-----



Exercise 5

Create a Trivia Quiz on something you like and are knowledgeable about.

Guidelines

1. Ask any amount of questions with user input.
2. Put the answer to all the questions in a function that will compute the score, and return a percentage. For example, if a user gets $\frac{3}{5}$ questions right, the function returns 0.6 or 60.00.
3. Put the IQ of any user into 3 categories:
 - a. Score ≥ 80 is high IQ
 - b. $40 \leq \text{Score} < 80$ is mid IQ
 - c. Score < 40 is low IQ

NOTE: 1 means TRUE and 0 means FALSE



Exercise 5

```
1 def getScore(a1,a2,a3,a4,a5):
2     sum = 0
3     if a1 == 1: sum += 1
4     if a2 == 1: sum += 1
5     if a3 == 0: sum += 1
6     if a4 == 0: sum += 1
7     if a5 == 0: sum += 1
8     return (sum/5)*100
```

```
10 a1 = int(input("There was no world cup in 1942 because of
    war. "))
11 a2 = int(input("Real Madrid won 14 Champions Leagues in
    its history. "))
12 a3 = int(input("Mbappe was the top scorer at the 2018
    world cup. "))
13 a4 = int(input("Ronaldo is the all time top scorer in La
    Liga. "))
14 a5 = int(input("The first world cup in history took place
    in 1926 and was won by Uruguay. "))
15
16 score = getScore(a1,a2,a3,a4,a5)
17
18 if score >= 80:
19     print(f"You have a high football IQ with a {score}%")
20 elif score >= 40 and score < 80:
21     print(f"You have mid football IQ with a {score}%")
22 else:
23     print(f"I doubt you are a soccer fan with a {score}%")
```