

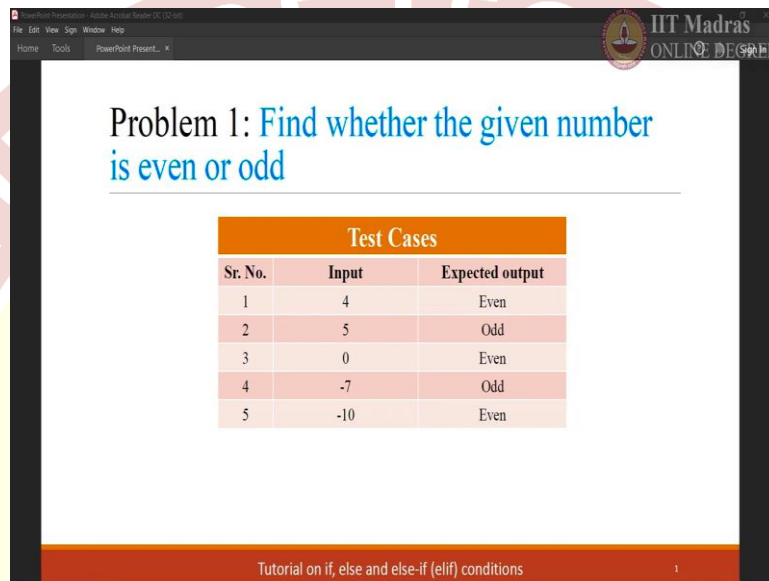


IIT Madras

ONLINE DEGREE

Programming in Python
Professor Sudarshan Iyengar
Department of Computer Science and Engineering
Indian Institute of Technology Ropar
Omkar Joshi
Course Instructor
Indian Institute of Technology Madras Online Degree Program
Tutorial on if, else and else-if (elif) conditions

(Refer Slide Time: 0:17)



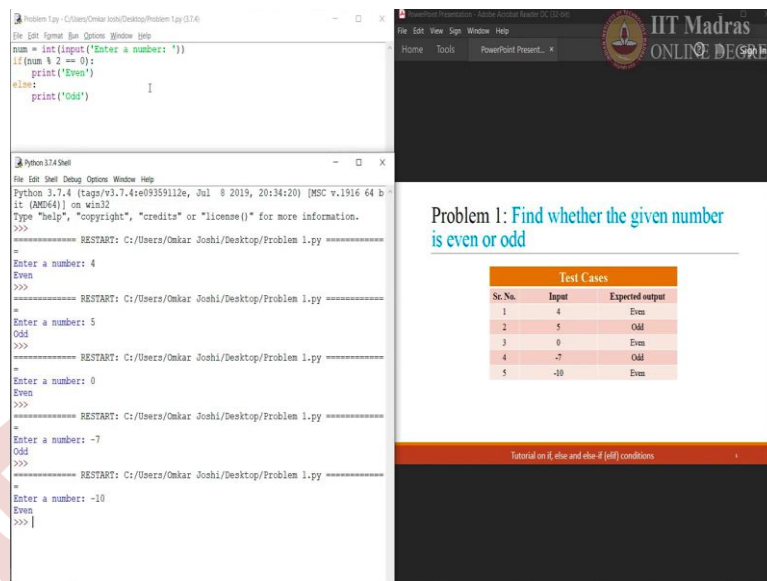
Problem 1: Find whether the given number is even or odd

Test Cases		
Sr. No.	Input	Expected output
1	4	Even
2	5	Odd
3	0	Even
4	-7	Odd
5	-10	Even

Tutorial on if, else and else-if (elif) conditions

Hello python students. In this tutorial we will see how to write python codes for some common examples based on real time applications using if, else and else-if conditions. Let us look at this first problem statement. Find whether the given number is even or odd. In addition to that some test cases are given. We are supposed to execute our python code using these given set of inputs and check whether we are getting the expected output or not. Now let us open a python editor and try to write python code which checks whether the given number is even or odd.

(Refer Slide Time: 1:06)



```
Problem 1.py - C:/Users/Omkar Joshi/Desktop/Problem 1.py (3.7.4)
File Edit Format Run Options Window Help
num = int(input("Enter a number: "))
if (num % 2 == 0):
    print('Even')
else:
    print('Odd')
```

```
Python 3.7.4 Shell
File Edit Shell Debug Options Window Help
Python 3.7.4 (tags/v3.7.4:09359112e, Jul 8 2019, 20:34:20) [MSC v.1916 64 b-
it (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/Omkar Joshi/Desktop/Problem 1.py =====
>>>
Enter a number: 4
Even
>>>
===== RESTART: C:/Users/Omkar Joshi/Desktop/Problem 1.py =====
>>>
Enter a number: 5
Odd
>>>
===== RESTART: C:/Users/Omkar Joshi/Desktop/Problem 1.py =====
>>>
Enter a number: 0
Even
>>>
===== RESTART: C:/Users/Omkar Joshi/Desktop/Problem 1.py =====
>>>
Enter a number: -7
Odd
>>>
===== RESTART: C:/Users/Omkar Joshi/Desktop/Problem 1.py =====
>>>
Enter a number: -10
Even
>>>]
```

IIT Madras
ONLINE DEGREE

Problem 1: Find whether the given number is even or odd

Test Cases		
Sl. No.	Input	Expected output
1	4	Even
2	5	Odd
3	0	Even
4	-7	Odd
5	-10	Even

Tutorial on if, else and else if (elif) conditions


As it says, the given number which means we have to accept some number from the user. In order to do so, we write one function called input, this particular message which says enter a number will be displayed when we execute this particular python code. Now this input can be of any data type, but as we can see here, in order to find even or odd, we require an integer. In order to convert entered input into integer, we have to write one function called int.

Now this particular will accept the number from user and convert it to integer, but for further processing we should store this number in to some variable. Now we have to check whether the given number is even or odd. As we know any number which is divisible by 2 is an even number, we will use this particular property to find whether the entered number is even or odd.

If $\text{num} \bmod 2$ is equal equal to 0, this particular line uses a modulo operator which returns remainder after division. If the remainder is 0, that means number is exactly divisible by 2 which means the number is even. Then the next part is for odd numbers, if this particular condition fails, then the number is odd, hence we have to write that in an else. Now let us try to execute this particular program using the given set of inputs.

Enter a number, first input is 4, even, that is an expected value. Let us try the next input 5, which is odd, 0 which is an even number, minus 7 is also odd and minus 10 is even. All the given test cases matches exactly with the outputs what we are getting through our python code which indicates that our python code is correct. Now lets us move on to next problem statement.

(Refer Slide Time: 4:00)




Problem 2: Find whether the given number ends with 0 or 5 or any other number

Test Cases		
Sr. No.	Input	Expected output
1	20	0
2	14	Other
3	5	5
4	0	0
5	-27	Other
6	-10	0

Tutorial on if, else and else-if (elif) conditions 2

Find whether the given number ends with 0 or 5 or any other number. This is little bit more trickier than the previous question. Here we have to check what is the last digit in the given number, whether its 0, 5 or any other number. Here also we can use that modulo operator to get the remainder. If we divide the given number by 10, then it will give us the remainder.

(Refer Slide Time: 4:35)



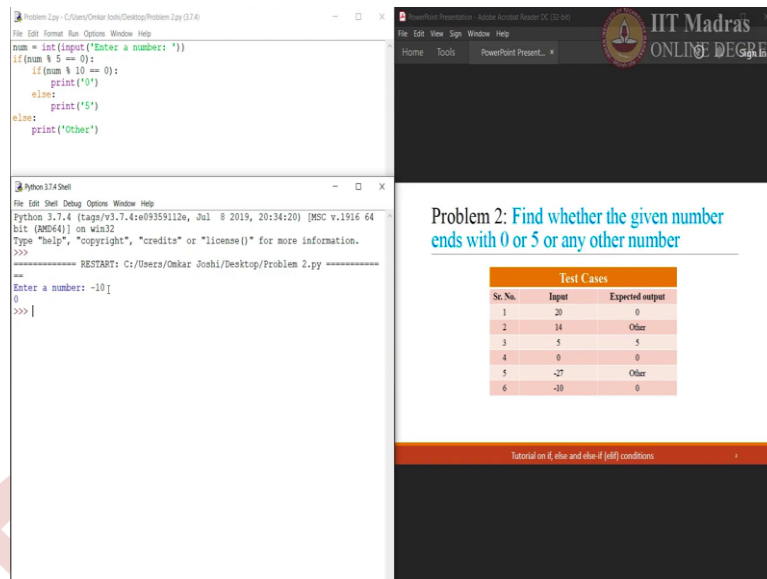
Problem 2: Find whether the given number ends with 0 or 5 or any other number

Test Cases		
Sr. No.	Input	Expected output
1	20	0
2	14	Other
3	5	5
4	0	0
5	-27	Other
6	-10	0

Tutorial on if, else and else-if (elif) conditions 3

```
num = int(input("Enter a number: "))
if (num % 5 == 0):
    if (num % 10 == 0):
        print("0")
    else:
        print("5")
else:
    print("Other")
```

```
Enter a number: 20
0
Enter a number: 14
Other
Enter a number: 5
5
Enter a number: 0
0
Enter a number: -27
Other
```



Here also we have to accept a number from the user and it has to be an input, hence we will declare a one variable called num, use int function for the input, this particular line will accept a number from the user as a input. Now we have to check the last digit of the entered number. The last digit of the given number will be 0, if it is divisible by 10 and 5, whereas the last digit will be 5, if it is divisible by only 5, hence we should start with that particular condition.

If $\text{num} \bmod 5$ is equal equal to 0, then we will do something or else we have to say that number is from other category. Now if the number is divisible by 5, then we have to check whether the number is divisible by only 5 or also 10. In order to capture that we have to add one more if block which checks whether the number is divisible by 10 or not. If number is divisible by 10, that means definitely the number ends with 0, hence we can print 0, otherwise we will print 5.

Now let us try to execute this particular code with the given test cases. First input it is 20, we are getting the expected output. Let us try the next input 14, other; 5, 5; 0, 0. Now let us try some negative numbers also, minus 27, it falls in other category; minus 10 gives 0. As all the given test cases are giving the expected output that means the written code is correct. Now let us move on to next problem statement.

(Refer Slide Time: 7:49)

Problem 3: Find the grade of student based on the given marks from 0 to 100.

Student grades		
Sr. No.	Grade	Marks range
1	A	≥ 90
2	B	≥ 80 and < 90
3	C	≥ 70 and < 80
4	D	≥ 60 and < 70
5	E	< 60

Test Cases		
Sr. No.	Input	Expected output
1	95	A
2	87	B
3	70	C
4	61	D
5	0	E
6	100	A
7	-5	Invalid input
8	110	Invalid input

Tutorial on if, else and else-if (elif) conditions

Find the grade of student based on the given marks from 0 to 100. The first table defines how this grades should be allotted; A grade means greater than or equal to 90, B grade falls under greater than equal to 80 and less than 90, C grade greater than equal to 70 and less than 80, and so on for grades D and E. Whereas the second table represent the test cases.

(Refer Slide Time: 8:24)

Problem 3: Find the grade of student based on the given marks from 0 to 100.

```
marks = int(input("Enter marks: "))
if(marks >= 90):
    print('A')
if(marks >= 80 and marks < 90):
    print('B')
if(marks >= 70 and marks < 80):
    print('C')
if(marks >= 60 and marks < 70):
    print('D')
if(marks < 60):
    print('E')
```

Enter marks: 95
A

Traceback (most recent call last):
File "C:/Users/Omkar Joshi/Desktop/Problem 3.py", line 1, in <module>
NameError: name 'marks' is not defined

Student grades		
Sr. No.	Grade	Marks range
1	A	≥ 90
2	B	≥ 80 and < 90
3	C	≥ 70 and < 80
4	D	≥ 60 and < 70
5	E	< 60

Test Cases		
Sr. No.	Input	Expected output
1	95	A
2	87	B
3	70	C
4	61	D
5	0	E
6	100	A
7	-5	Invalid input
8	110	Invalid input

Tutorial on if, else and else-if (elif) conditions

The collage consists of four screenshots. The top-left shows the Python script for Problem 3. The top-right shows a presentation slide with the problem statement and test cases. The bottom-left shows the script being executed in a Python 3.7.4 Shell. The bottom-right shows the presentation slide again.

Problem 3: Find the grade of student based on the given marks from 0 to 100.

Student grades			Test Cases		
St. No.	Grade	Marks range	St. No.	Input	Expected output
1	A	>=90	1	95	A
2	B	>=80 and <90	2	87	B
3	C	>=70 and <80	3	70	C
4	D	>=60 and <70	4	61	D
5	E	<60	5	0	E
			6	100	A
			7	-5	Invalid input
			8	110	Invalid input

Python 3.7.4 Shell

```

marks = int(input("Enter marks: "))
if(marks >= 90):
    print("A")
elif(marks >= 80 and marks < 90):
    print("B")
elif(marks >= 70 and marks < 80):
    print("C")
elif(marks >= 60 and marks < 70):
    print("D")
elif(marks < 60):
    print("E")

```

Execution Log:

```

>>>
Enter marks: 95
A
NameError: name 'marks' is not defined
>>>
Enter marks: 87
B
>>>
Enter marks: 70
C
>>>
Enter marks: 61
D
>>>
Enter marks: 0
E
>>>
Enter marks: 100
A
>>>
Enter marks: -5
E
>>>

```

Now let us write the python code for this third problem statement. Marks is equals to int of input of, enter marks. Now we have to check whether the entered marks are greater than equal to 90 or not. If that is the case, we will print grade as A. Let us write that in terms of if block, if marks greater than equal to 90, then print A.

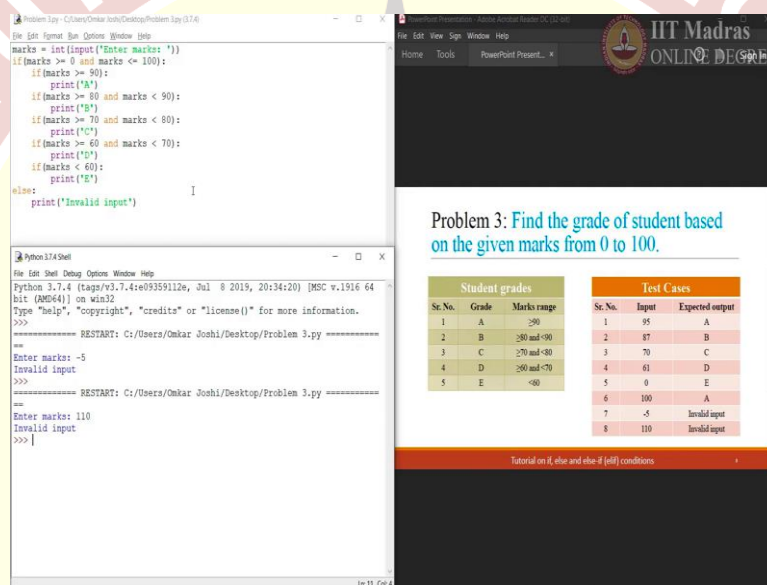
If that is not the case we have to check another condition which checks whether the marks are greater than equal to 80 and marks are less than 90, then print B. Similar if block we have to add for grade C, D and E. Marks greater than equal to 70 and marks less than 80, then we will print grade C. Now as all the grades are written in terms of if blocks, let us try to execute the program using the given test cases.

First input marks 95, here we are getting an error which says name error, name marks is not defined. Now this is happening, because we have a made a typing mistake while using the

variable name marks. Over here you can see even the line number is mentioned which is line number 6 which is this particular line. Let us correct the mistake and try to execute the code again.

Marks 95 and the grade is A, which is correct. Next 87, that is B; 70 it is C; 61 is D; 0 is E; 100 is A; minus 5, we are getting output as E, but the expected output says invalid input. Now this is the place where we have missed something, because as the problem statement says find the grade of students based on the given marks from 0 to 100. Which means any number smaller than 0 or greater than 100 should be termed as invalid input.

(Refer Slide Time: 12:22)



```
marks = int(input("Enter marks: "))
if(marks >= 0 and marks <= 100):
    if(marks >= 90):
        print("A")
    if(marks >= 80 and marks < 90):
        print("B")
    if(marks >= 70 and marks < 80):
        print("C")
    if(marks >= 60 and marks < 70):
        print("D")
    if(marks < 60):
        print("E")
else:
    print('Invalid input')
```

Python 3.7.4 Shell

```
Python 3.7.4 (tags/v3.7.4:09359112e, Jul 8 2019, 20:34:20) [MSC v.1916 64-bit (AMD64)] on win32
Type "help()", "copyright()", "credits()" or "license()" for more information.
>>>
===== RESTART: C:/Users/Omkar Joshi/Desktop/Problem 3.py =====
Enter marks: -5
Invalid input
>>>
===== RESTART: C:/Users/Omkar Joshi/Desktop/Problem 3.py =====
Enter marks: 110
Invalid input
>>> |
```

Problem 3: Find the grade of student based on the given marks from 0 to 100.

Student grades			Test Cases		
Sl. No.	Grade	Marks range	Sl. No.	Input	Expected output
1	A	>= 90	1	95	A
2	B	>= 80 and < 90	2	87	B
3	C	>= 70 and < 80	3	70	C
4	D	>= 60 and < 70	4	61	D
5	E	< 60	5	0	E
			6	100	A
			7	-5	Invalid input
			8	110	Invalid input

Tutorial on if, else and else if (elif) conditions

Now in order to accommodate this particular condition where the marks should be entered in the range of 0 to 100, we should check all these if conditions currently available in the code only when the marks value is in between 0 to 100. That means if marks greater than equal to 0 and marks less than equal to 100, then only we should execute all these conditions.

Otherwise, we will print invalid input. Now let us try this modified python code with those test cases once again. Let us try to enter minus 5 again, invalid input as expected; let us try some higher number also, 110 again invalid input. Now all our conditions are correct. Now as all our test cases are giving us expected output which means now this code is correct.

(Refer Slide Time: 13:57)

Problem 3: Find the grade of student based on the given marks from 0 to 100.

St. No.	Grade	Marks range
1	A	>=90
2	B	>=80 and <90
3	C	>=70 and <80
4	D	>=60 and <70
5	E	<60

St. No.	Input	Expected output
1	95	A
2	87	B
3	70	C
4	61	D
5	0	E
6	100	A
7	-5	Invalid input
8	110	Invalid input

Tutorial on if, else and else if (elif) conditions

```
marks = int(input("Enter marks: "))
if(marks >= 0 and marks <= 100):
    if(marks >= 90):
        print("A")
    elif(marks >= 80):
        print("B")
    elif(marks >= 70):
        print("C")
    elif(marks >= 60):
        print("D")
    else:
        print("E")
else:
    print("Invalid input")
```

```
>>>
===== RESTART: C:/Users/Omkar Joshi/Desktop/Problem 3.py =====
>>>
Enter marks: 61
D
>>>
===== RESTART: C:/Users/Omkar Joshi/Desktop/Problem 3.py =====
>>>
Enter marks: 0
E
>>>
===== RESTART: C:/Users/Omkar Joshi/Desktop/Problem 3.py =====
>>>
Enter marks: 100
A
>>>
===== RESTART: C:/Users/Omkar Joshi/Desktop/Problem 3.py =====
>>>
Enter marks: -5
Invalid input
>>>
===== RESTART: C:/Users/Omkar Joshi/Desktop/Problem 3.py =====
>>>
Enter marks: 110
Invalid input
>>>
```

As I mentioned earlier, in this tutorial, we will see one more concept called else-if. Let us try to introduce that else-if concept with this particular problem statement. As you can see over here, every condition inside first if block requires an AND operation of two different conditions. This particular thing is required to make sure all those students who are counted for A grade, does not appear again in B grade. This particular thing can be achieved using a different kind of conditional statement called else-if, where the second block gets executed only if first block does not satisfy the given condition.

Let us try to convert this code using else-if block. else-if, when we add that else-if, this particular part becomes redundant, hence it can be removed. Similar thing can be done over

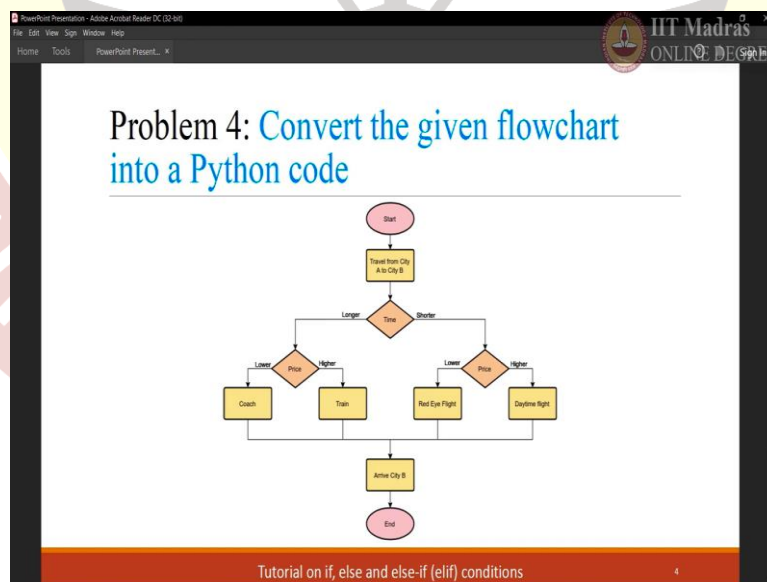
here and here. As we have using this if else-if else-if, we should convert the last code also as simply else. Now let us try to analyze this updated version of the python code.

If checks whether the marks are greater than 90, if that is the case it will print A, if not, which is else-if marks greater than equal to 80. When this particular condition gets executed which itself means the marks are not greater than 90, hence we do not have to execute that additional part which we had earlier.

Similar thing will happen when we execute this particular condition marks greater than equal to 70 gets executed only when the entered marks are less than 80 and so on. At the end, whenever we reach to this else statement that means the entered marks are less than 60, hence we are simply printing E, instead of checking in any additional condition. Now let us try to execute the given code with all set of test cases.

First the input 95 A grade, 87 B, 70 C, 61 D, 0 E, 100 A, minus 5 invalid, 110 invalid; all test cases are successfully giving the expected outputs, which means the code is correct. This transition from previous python code to this updated python code, simplifies the condition checks for each if block. Three additional conditions we have removed and this particular thing optimizes the entire code. Now let us move on to next problem statement.

(Refer Slide Time: 18:32)



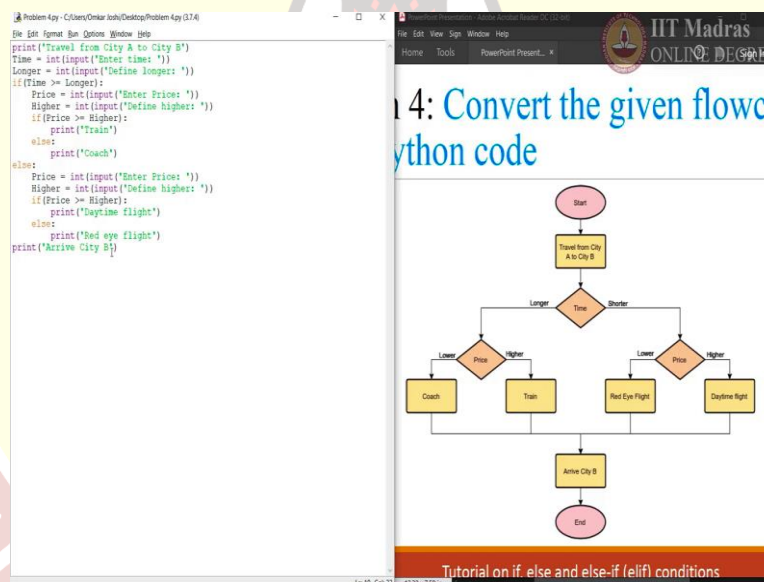
Problem 4; convert the given flowchart into a python code. Now this is a different kind of a problem statement. Here instead of test cases, a sample flowchart is given and we have to convert this flowchart into its equivalent python code. All those yellow blocks represent that

we have to print the given statement, whereas we have those three if conditions where we have to take some decision based on the values.

Which means we have to accept the user input for time and those two price if blocks. Whenever we get that time input from the user, we also have to determine, whether the entered time by the user is considered to be longer or shorter. As the words longer and shorter are relative terms, we also have to ask user what exactly the word longer signifies over here.

Because for one person longer may have a different value, whereas for different user it will have a different value, hence it is better to keep that as a variable and accept that as well from the user. Similar thing is required for price if condition, where the prices higher and lower has to be accepted from the user. Now let us try to write a python code which represents this particular flowchart.

(Refer Slide Time: 20:31)



Start first statement print; travel from city A to city B. Next we have to check whether the time is longer or shorter. In order to do that, first we have to accept the value of time, time is equal to int input enter time. Once we have this time variable's value accepted from user, we also have to setup the meaning of word longer and shorter. It can be done using another input which accepts the value for longer.

If we know the meaning of longer then we can simply derive the meaning of shorter from it, hence we do not have to accept the value against shorter. Now let us try to setup that if block using these two variables. If time is greater than equal to longer, then we have to check price.

Once again we have to accept user inputs for price and higher. Price is equal to int of input of enter price, higher is equal to int of input of define higher.

Once we have these two input values defined, then we can setup that if block which checks whether to choose a train or a coach. If price is greater than equal to higher, then we will opt for train, else we will opt for coach. Similar thing we have to do for the right hand side part of the pseudo code, which is the else part of the initial if condition we checked for the time.

Else again we will accept price and higher from the user. Based on the values of these two variables, we will decide to choose whether red eye or a day time flight is better option, if price is greater than equal to higher, print day time flight, else print red eye flight. After this, irrespective of execution of if side path or else side path, which means irrespective of time being longer or shorter and then prices being higher or longer, we will reach to city B, which is indicated using that last print statement arrive city B.

Hence, we will print that statement outside both this particular if else blocks. As no test cases are given over here, we do not have to test this particular code against specific inputs. If we read the key points from the written python code, it will exactly represent the given flowchart.

Travel from city A to B, if time is longer and price is higher, go for a train; if price is lower, go for coach; if time is shorter and price is higher, go for day time flight; if price is shorter, go for red eye flight, irrespective of all this arrive at city B. Thank you for watching this tutorial. Happy learning!