C5 – S1 – THEORY

**EXERCISE 1**

**PROBLEM:**

* Enter a number.
* Check if this number is in one of the bellow ranges:

1 to 10

29 to 51

76 to 101

* Print True if the number is in one of the ranges, print False otherwise.

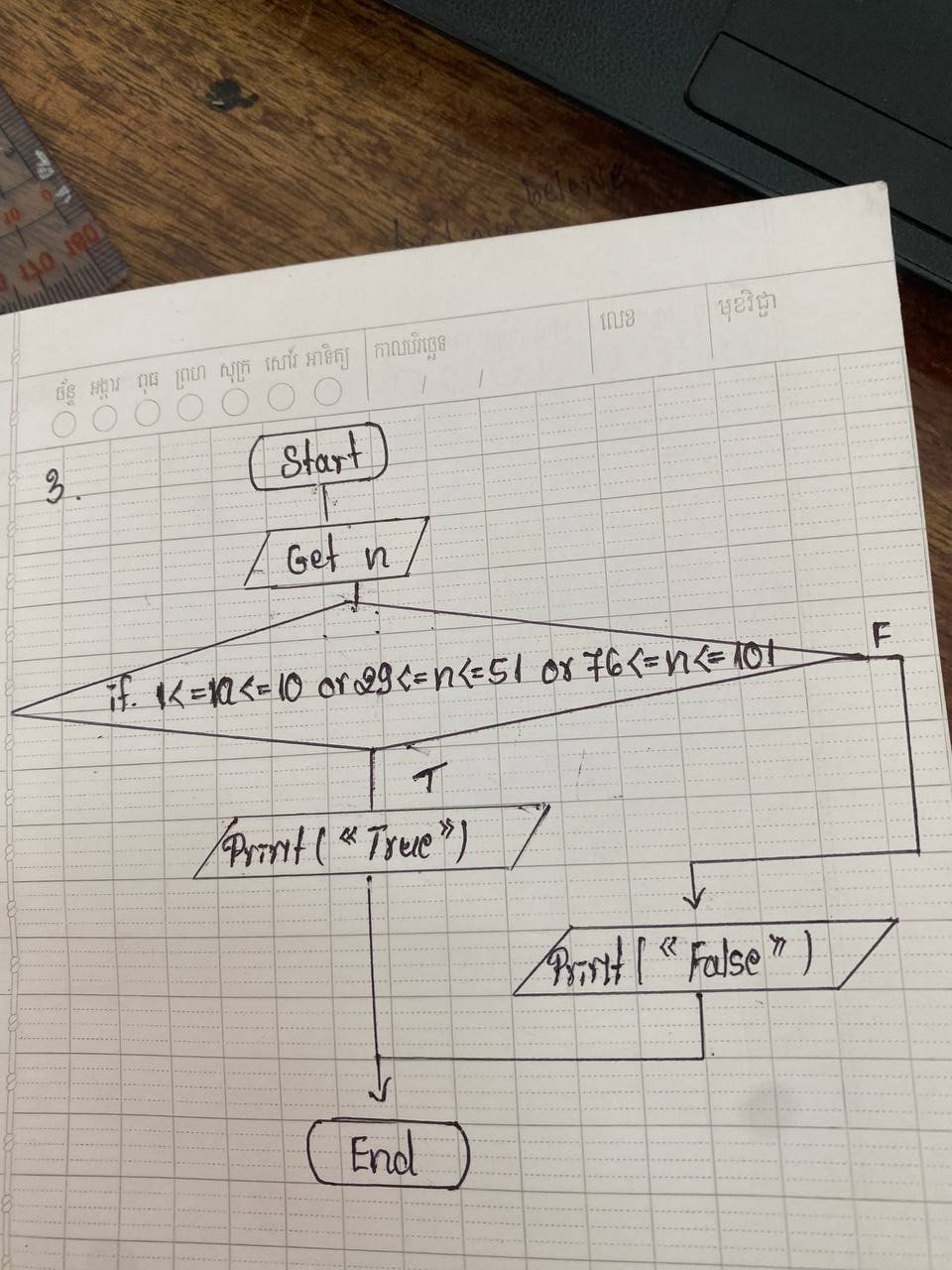
**Q1** – Complete the missing outputs

|  |  |
| --- | --- |
| **INPUT** | **OUTPUT** |
| 11 | False |
| 50 | True |
| 88 | True |
| 30 | True |
| 101 | True |

**Q2** – Analyze the symbols you need to solve this problem

|  |  |  |
| --- | --- | --- |
| Element | Do you need it? | For what? |
| Action | no |  |
| Decision | yes | For store condition |
| Repeat | no |  |
| Input / Output | yes | For input and output |

**Q3** – Create a flowchart to solve this problem.



**Q4** – **Execute** the flowchart: What is the result of your flowchart with those inputs?

|  |  |
| --- | --- |
| **INPUT** | **OUTPUT** |
| -1 | False |
| 0 | False |
| 8 | True |
| 11 | False |
| 15 | False |
| 29 | True |
| 35 | True |
| 75 | False |
| 80 | True |
| 110 | Fase |

**Q5** – Review the code and find **the error** and explain them.

error, because it used a lot of condition but we need only one condition.

# Check if a number is in one of the 3 ranges: 1 to 10 or 29 to 51 or 76 to 101

value = int(input())

inRange = False

if value >= 1 or value <= 10:

inRange = True

elif value >= 29 or value <= 51:

inRange = True

elif value >= 76 or value <= 101:

inRange = True

print(inRange)

**Q6** If this code is a valid code? Explain why

if we want to this code display valid we need to add else: inRange=true because all of this code display false only all of condition.

# Check if a number is in one of the 3 ranges: 1 to 10 or 29 to 51 or 76 to 101

value = int(input())

inRange = True

if value < 1:

inRange = False

elif value > 10 and value < 29:

inRange = False

elif value > 51 and value < 76:

inRange = False

elif value > 101:

inRange = False

print(inRange)

**Q7**– Write your own good code to solve this problem.

This time, you can us **1 condition** only

n= input()

if 1<=n<=10 or 29<=n<=51 or 76<=n<=101:

print(“True”)

else:

print(“False”)

# EXERCICE 2

* Input a text in the console.
* Print the number of points related to this word, following the below rules.

|  |  |
| --- | --- |
| **IF THE WORD CONTAINS** | **THEN THE POINTS ARE** |
| **One** ‘*A’* **or more** | 10 points |
| **One ‘***B’* **or more** | 20 points |

* Note: you can cumulate the rules: if you have some “A” and some “B” it will be 10+20 = 30 points!
* If no rules match, then the result is 0 points.

***Examples***

|  |  |
| --- | --- |
| **INPUT** | **OUTPUT** |
| KKK | 0 |
| BCA | 30  *Because we found 1 ‘B’ and 1 ‘A’: 10 + 20* |
| MMBBR | 20 |
| MAARTDAC | 10 |
| AABBBB | 30 |
| C | 0 |

**Q1** – What will be the **results** for those inputs?

|  |  |
| --- | --- |
| INPUT | OUTPUT |
| DADADA | 10 |
| ACAAAB | 30 |
| AAAAAA | 10 |
| QWERTY | 0 |

**Q2** – **Fill up the gap** on this flowchart.

If n[i]==”B” :

If n[i]==”A” :

I<len(PIONT)

Set POINT\_A to 0

Get *WORD*

Yes

Set POINT\_B to 0

No

Done

Set POINT\_A to 10

TEST

Yes

Set POINT\_B to 20

No

Display POINT\_A + POINT\_B

END

**Q3** – Implement it and test it with the inputs of the first question.

sum1=0

sum2=0

n=input()

for i in range(len(n)):

    if n[i]=="A" or n[i]=="a":

        sum1=10

    elif n[i]=="B" or n[i]=="b":

        sum2=20

print(sum1+sum2)

EXERCICE 3

* Execute mentally the below code and write, for each step of execution the value of each variable.
* If the variable is not defined yet, write “?”

a = "roman"

b = a[2]

c = a + b

a = c[-1]

|  |  |  |  |
| --- | --- | --- | --- |
| STEP | A | B | C |
| 1 | “roman” |  |  |
| 2 | “roman” | “m” |  |
| 3 | “roman” | “m” | “romanm” |
| 4 | “m” | “m” | “romanm” |