

0.0.1 ICS 104 - Introduction to Programming in Python and C

1 Programming with numbers and Strings 1

1.1 Lab Learning Outcomes

- Apply problem solving in program development
- Comprehend variables and types
- Write python expressions

1.2 Problem Solving

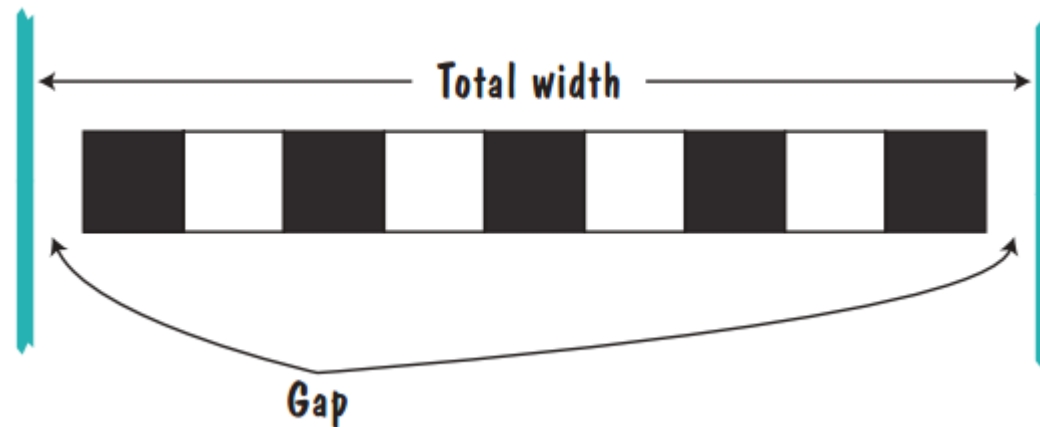
- One technique to apply problem solving is to do the following
 - Read the problem statement and understand it fully.
 - Determine **what** needs to be done
 - The available **input** and the requested result (**output**).
 - The sequence of **steps** to be carried out before we can produce the **output** .
 - Don't worry about how to do it for now.
 - Develop and describe the **algorithm** in pseudo-code (**how** to carry out the above **steps**).
 - An algorithm is a **sequence** of steps that is **unambiguous**, **executable**, and **terminating**.
 - Test the algorithm with simple inputs
 - to have more confidence that it is correct.
 - Translate the pseudo code into a Python program.
 - Compile and test your program.

1.3 Example (Section 2.3 from the textbook)

- A row of black and white tiles needs to be placed along a wall. For aesthetic reasons, the architect has specified that the first and last tile shall be black.

- Your task is to compute the number of tiles needed and the gap at each end, given the space available and the width of each tile.

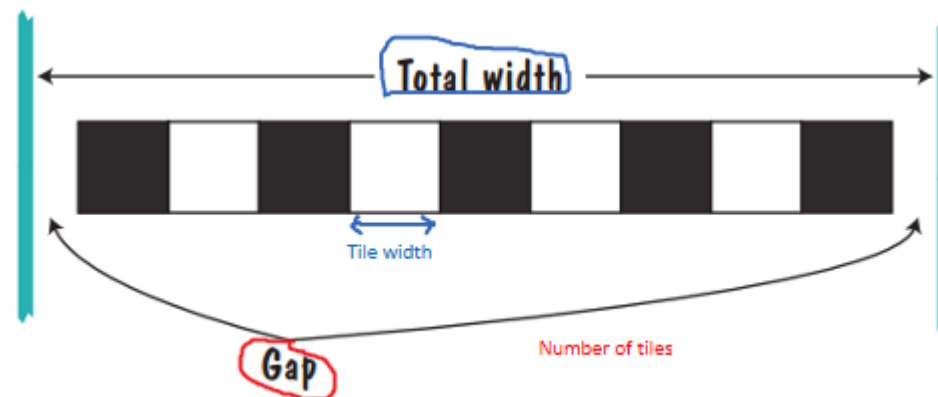
- The problem can be depicted as follows:



- What is the input and what is the output?

In []:

1



In [1]:

```

1 #totalWidth=10
2 #tileWidth=1
3
4

```

- Input
 - Total width
 - Tile width
- Output
 - Number of tiles
 - Gap between the first/last tile and the wall

- What do we need to do to find the output?



- Since we know that the first tile is always black, the number of tiles left is a multiple of 2 (white and black). Hence, we need to do the following
 - Find the total width after removing the length of the black tile.
 - Find how many tiles (in pairs) that can fit the remaining space.
 - The number of tiles needed is one plus the number of tiles in the previous step.
 - Find the wall gap after using the number of tiles found.

- Following is the pseudo code for the above steps
 - $\text{widthWithoutBlackTile} = \text{Total Width} - \text{Tile Width}$
 - Number of white-black tile pairs = the integer part of
$$\frac{\text{widthWithoutBlackTile}}{(2 * \text{TileWidth})}$$
 - Number of Tiles = $1 + 2 * \text{Number of white-black tile pairs}$
 - Length of used tiles = Number of Tiles * Tile width
 - Total gap = Total Width - Length of used tiles
 - Gap at each end = Total gap / 2

- Apply the steps on a wall of width 10 and a tile of width 1.

- $\text{widthWithoutBlackTile} = \text{Total Width} - \text{Tile Width}$
- Number of white-black tile pairs = the integer part of
$$\frac{\text{widthWithoutBlackTile}}{(2 * \text{TileWidth})}$$
- Number of Tiles = $1 + 2 * \text{Number of white-black tile pairs}$
- Length of used tiles = Number of Tiles * Tile width
- Total gap = Total Width - Length of used tiles
- Gap at each end = Total gap / 2

- $\text{widthWithoutBlackTile} = 10 - 1 = 9$
- Number of white-black tile pairs = the integer part of
$$(9/2) = 4$$
- Number of Tiles =
$$1 + (2 * 4) = 9$$
- Length of used tiles =
$$9 * 1 = 9$$
- Total gap =
$$10 - 9 = 1$$
- Gap at each end =
$$1/2 = 0.5$$

1.4 Exercises

1.4.1 Exercise 1

- Convert the following pseudo code into python. Run your program for different input values.
 - $\text{widthWithoutBlackTile} = \text{Total Width} - \text{Tile Width}$
 - Number of white-black tile pairs = the integer part of
$$\frac{\text{widthWithoutBlackTile}}{(2 * \text{TileWidth})}$$
 - Number of Tiles = $1 + 2 * \text{Number of white-black tile pairs}$
 - Length of used tiles = Number of Tiles * Tile width
 - Total gap = Total Width - Length of used tiles
 - Gap at each end = Total gap / 2

Sample run when:

TotalWidth = 23

TileWidth = 2

Number of tiles = 11

Gap on two sides = 0.5

Number of Black Tiles 6

Number of White Tiles 5

In [7]:

```

1  ###%writefile lab02Ex1_YourID.py
2  # NOTE: uncomment the above line after you finish writing, running and testing your code.
3
4  totalwidth = 23
5  tilewidth = 2
6  print("Total width = ", totalwidth)
7  print("Tile width = ", tilewidth)
8
9  widthwithoutblacktile = totalwidth - tilewidth
10 pairs = widthwithoutblacktile // (2 * tilewidth)
11 numberoftiles = 1 + 2 * pairs
12 print("Number of tiles =", numberoftiles )
13
14 lengthofusedtiles = numberoftiles * tilewidth
15 totalgap = totalwidth - lengthofusedtiles
16 gapateachend = totalgap / 2
17 print("Gap on two sides = ", gapateachend)
18 print("Number of Black Tiles = ", pairs + 1)
19 print("Number of white Tiles = ", pairs )

```

Total width = 23

Tile width = 2

Number of tiles = 11

Gap on two sides = 0.5

Number of Black Tiles = 6

Number of wight Tiles = 5

1.4.2 Exercise 2

- Suppose the architect specifies a pattern with black, gray, and white tiles, like this:



Again, the first and last tile should be black. Solve this problem by modifying the pseudo code of the previous exercise.

- Include your pseudo-code here
 - Step 1
 - Step 2

- ...

Sample run when:

TotalWidth = 26

TileWidth = 1

Number of tiles = 25

Gap on two sides = 0.5

Nuber of Black Tiles 7

Nuber of White Tiles 6

Nuber of Gray Tiles 12

In [9]:

```
1  ###writefile lab02Ex2_YourID.py
2  # NOTE: uncomment the above line after you finish writing, running and testing your code.
3
4  totalwidth = 26
5  tilewidth = 1
6  print("Total width = ", totalwidth)
7  print("Tile width = ", tilewidth)
8
9  widthwithoutblacktile = totalwidth - tilewidth
10 pairs = widthwithoutblacktile // (4 * tilewidth)
11 numberoftiles = 1 + 4 * pairs
12 print("Number of tiles =", numberoftiles )
13
14 lengthofusedtiles = numberoftiles * tilewidth
15 totalgap = totalwidth - lengthofusedtiles
16 gapateachend = totalgap / 2
17 print("Gap on two sides = ", gapateachend)
18 print("Number of Black Tiles = ", pairs + 1)
19 print("Number of White Tiles = ", pairs )
20 print("Number of Gray Tiles = ", 2 * pairs )
```

Total width = 26

Tile width = 1

Number of tiles = 25

Gap on two sides = 0.5

Number of Black Tiles = 7

Number of White Tiles = 6

Number of Gray Tiles = 12

1.4.3 Exercise 3

- Write a python code that calculates and prints the dot product of two vectors in the xy-plane.
- Initialize x1, y1, x2, and y2 with some values, representing the two vectors : $\langle x1, y1 \rangle$ and $\langle x2, y2 \rangle$
- Then Calculate and print the dot product.
- dot product= $x1*x2+y1*y2$

In [17]:

```
1  ###%writefile lab02Ex3_YourID.py
2  # NOTE: uncomment the above line after you finish writing, running and testing your code.
3
4  x1= 5
5  x2= 6
6  y1= 3
7  y2= 2
8
9  print("Vector1 =" , "<" , x1 , "," , y1 , ">")
10 print("Vector2 =" , "<" , x2 , "," , y2 , ">")
11 dotproduct = x1*x2 + y1*y2
12 print("dot product = " , dotproduct )
13
14
15
```

Vector1 = < 5 , 3 >

Vector2 = < 6 , 2 >

dot product = 36