LABORATOIRE
D'INFORMATIQUE
INTELLIGENTE
BISKRA

جامعة محمد خيضر بسكرة
MOHAMED KHIDER UNIVERSITY OF BISKRA

COMPUTER SCIENCE 1

DR. ADEL ABDELLI

adel.abdelli@univ-biskra.dz

2024/2025

CHAPTER 3

INDEXED VARIABLES

DR. ADEL ABDELLI

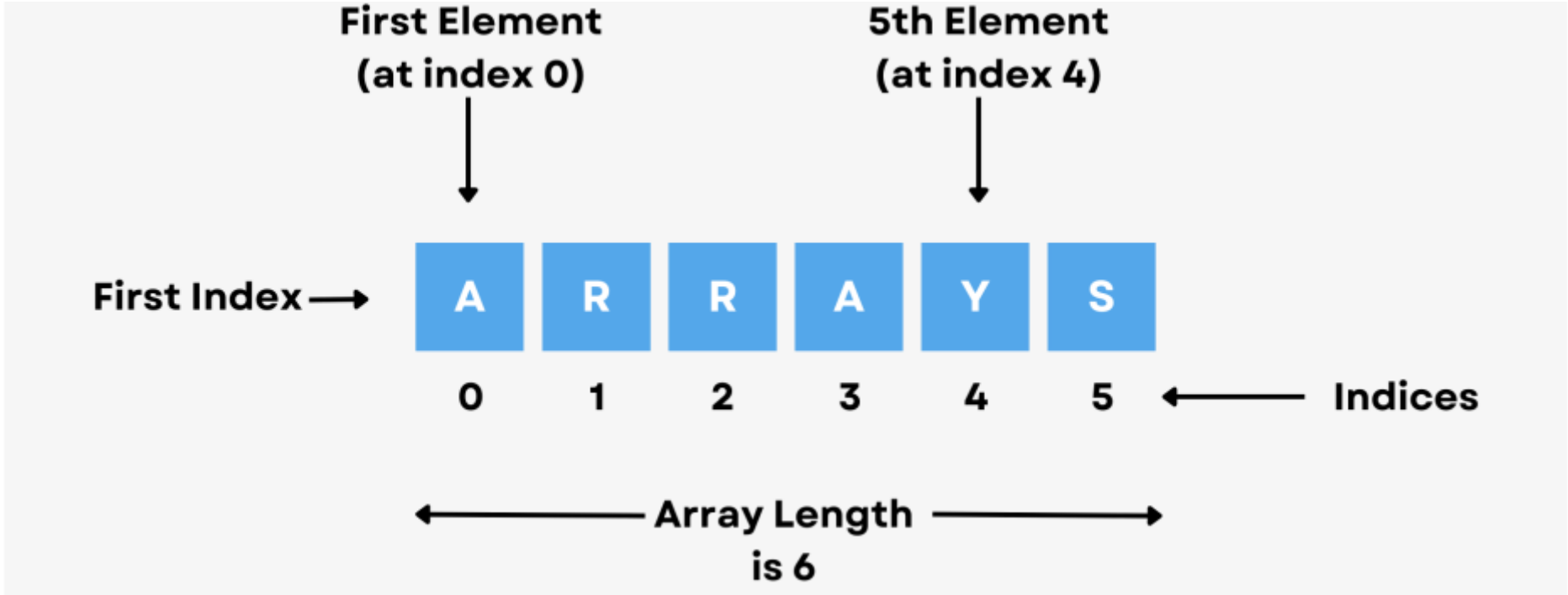
01/03/2025

1. Introduction

Indexed variables play a crucial role in computer programming and data structures, providing a way to manage and access multiple data elements using a single variable name. These variables are often associated with arrays, lists, or other data structures where elements are organized in a sequential order and accessed using an index. In this chapter, we are going to present arrays (one-dimensional/two-dimensional).

2. One-Dimensional Arrays

Arrays are used to represent a finite sequence of elements of the same type through a single variable. These elements can be integers, real numbers, strings, etc. They are stored in the different cells of the array, usually numbered from 1 to n, where n is the size of the array.



2.1 Declaration of an array:

The declaration of an array involves specifying its type, identifier, and size.

Identifier: Is the name of the array.
Size: Is the number of cells in the array.
Type: Is the type of elements: real, integer, character, boolean, or a compound type.

2.1.1 Syntax in Algorithm

```
Algorithm Test
 Var arrayName: array[lowerBound..upperBound] of element_type
Begin

End
```

Example:

```
T: array[0..9] of integer
V: array[0..100] of real
C: array[0..50] of character
```

2.1.2 Syntax in Python

array_name = []

Example:

```
T = []
T = [2, 4, 6, 7, 8]
```

Notes:

1. To access an element of the array, simply specify the index of the cell containing that element within square brackets. For example, to access the seventh element of the integer array above, write:

   t[6]

1. The following instruction assigns to the variable x the value of the first element of the array, which is 2:

   x ← t[1]

1. The designated element of the array can be used like any other variable: t[6] ← 4. This instruction modifies the seventh element of the array t.

2.2 Reading an Array:

2.2.1 Syntax in Algorithm

```
Algorithm readArray
Var T: array[0..n] of integer
      n, i : Integer
Begin
      Write('enter the size of the array')
      read(n)
      for i from 0 to n do
          read(t[i])
      End For
End
```

2.2 Reading an Array:

2.2.2 Syntax in Python

```
T = []
n = int(input("enter the size of the array"))
for i in range (0, n):
    x = int(input("enter a number"))
    T.append(x)
```

2.3 Displaying an Array:

2.3.1 Syntax in Algorithm

```
Algorithm showArray
Var T: array[0..n] of integer
    n, i: Integer
Begin
    for i from 0 to n do
        write(t[i])
    End For
End
```

2.3 Displaying an Array:
2.3.2 Syntax in Python

```
T = [1,2,3,4,5]

n = 5

for i in range(0, n):

        print(T[i])
```

Example:

Write a Python and an Algorithm that calculates the number of positive elements in an array of 10 elements.

```
n = 10
c = 0
T = []
print("enter the elements of the array")
for i in range(n):
    e = int(input("enter an element"))
    T.append(e)
print(T)
for i in range(n):
    if T[i] > 0:
        c = c+1

print("the total number of positive elements is:",c)
```

Example:

Write a Python and an Algorithm that calculates the number of positive elements in an array of 10 elements.

```
Algorithm positiveNumber
Const n=10
Var T: Array [0..n] of integer
x, i: integer
Begin
    Write ('Enter the elements ofthe array: ')
    For i from 0 to n do
        Read (T[i])
    EndFor
    x <- 0
    For i from 0 to n do
        If T[i] > 0 then
            x <- x + 1
        EndIf
    EndFor
    Write ('The number of positive elements=', x)
End
```

1D ARRAYS

DR. ADEL ABDELLI

01/03/2025

- **Showing element of an Array:**

Let's say we have the following array:

Arr = [1, 2, 3, 4, 5, 6, 7, 8]

➢ To show the 1st element we write the following line:

```
print(Arr[0])
```

➢ To show the last element we write one of the following line:

```
print(Arr[7])

print(Arr[-1])

length = len(Arr)
Print(Arr[length-1])
```

- **Showing element of an Array:**

➢ To show all the element of the array one by one (from the first to the last) we do the following:

```
Arr = [1, 2, 3, 4, 5, 6, 7, 8]

for i in range( len( Arr ) ):

   print( Arr[i] )
```

➢ Or from the last to the first:

```
Arr = [1, 2, 3, 4, 5, 6, 7, 8]

F_index = len(Arr) - 1

for i in range(F_index,-1,-1):

   print( Arr[i] )
```

- **Adding an elements to an Array:**

➢ To add an element to an array we use the function append as following:

```
Arr = [1, 2, 3, 4, 5, 6, 7, 8]
Arr.append(9)
Arr.append(10)
print(Arr)          # Output: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
```

Note: The append function add elements at the end of the array,

➢ Or by using insert function as following

```
Arr = [1, 2, 3, 4, 5, 6, 7, 8]
Arr.insert(0, -5)
Arr.insert(1,0)
print(Arr)          # Output: [-5, 0, 1, 2, 3, 4, 5, 6, 7, 8]
```

- **Removing an elements to an Array:**

➢ To remove an element from an array there is two methods:

1. By using pop() function: where it delete an element by giving its index:

```
Arr = [1, 2, 3, 4, 5, 6, 7, 8]

Arr.pop(3)

print(Arr) # Output: [1, 2, 3, 5, 6, 7, 8]
```

2. By using remove function: where it delete an element by giving its value:

```
Arr = [1, 2, 3, 4, 5, 6, 7, 8]

Arr.remove(5)

print(Arr) # Output: [1, 2, 3, 4, 6, 7, 8]
```

- **Changing a value of an elements in an Array:**

➢ To change the value of an element in array we do the following:

Exemple 1:

```
Arr = [1, 2, 3, 4, 5, 6, 7, 8]
Arr[0] = 15
print(Arr) # Output: [15, 2, 3, 4, 5, 6, 7, 8]
```

Exemple 2:

```
Arr = [1, 2, 3, 4, 5, 6, 7, 8]
Arr[-1] = 9
print(Arr) # Output: [1, 2, 3, 4, 5, 6, 7, 9]
```

- Exercise:

From the following array A, try to create another array B which contain only the odd numbers from the A array:

```
A = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
B = []
for i in range(len(A)):
    if A[i] % 2 == 1:
        B.append(A[i])
print(B)
```

- **Extending an Arrays:**

➢ Let's say we have two arrays A and B and we want to create new array C which is the concatenation of A and B:

```
A = [1, 2, 3, 4, 5, 6, 7, 8]
B = [9,10,11]
C = A + B
print(C)   # Output: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11]
```

➢ Let's say we have two arrays A and B and we want to extend A by adding B to it:

```
A = [1, 2, 3, 4, 5, 6, 7, 8]
B = [9,10,11]
A.extend(B)
print(A) # Output: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11]
```

- **Creating an array by asking the user the data:**

➢ To create an array by data entered from the user we do the following:

Exemple1:

```
A = []

n = int(input("enter the number of your siblings"))

for i in range(n):

   f = input("enter your sibling's name")

   A.append(f)

print(A)
```

- **Creating an array by asking the user the data:**

➢ To create an array by data entered from the user we do the following:

Exemple2:

```
A = []

for i in range(5):

   a = int(input("enter a number"))

   A.append(a)

print(A)
```

- **Slicing an Array:**

Sometimes we need only a part of the array, so we do slicing as following:

a[start:stop] # items begin from start until stop -1

a[start:] # items begin from start until the end of the array

a[:stop] # items from the beginning to stop-1

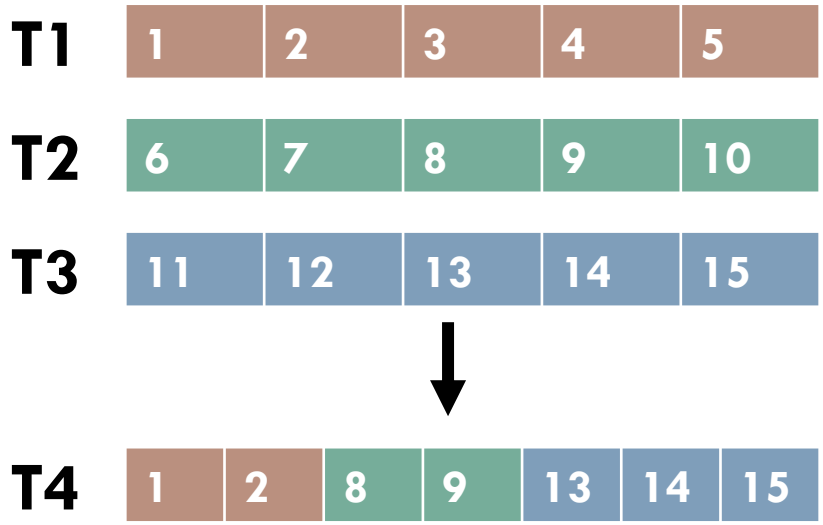
Exemples:

A = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11]

```
print( A[2:6] )    # Output: [3, 4, 5, 6]
print( A[2:] )     # Output: [3, 4, 5, 6, 7, 8, 9, 10, 11]
print( A[:6] )     # Output: [1, 2, 3, 4, 5, 6]
print( A[:] )      # Output: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11]
```

- ***Exercise 1:***

We have three arrays, T1, T2, T3 and we want to create an array T4 as following (using slicing and extend methods):



Solution:

```
T1 = [1,2,3,4,5]
T2 = [6,7,8,9,10]
T3 = [11,12,13,14,15]
T4 = T1[0:2] + T2[2:4] + T3[2:]
print(T4)
```

Exercise 2:

Write a Python code that find the maximum of a given array,

Solution:

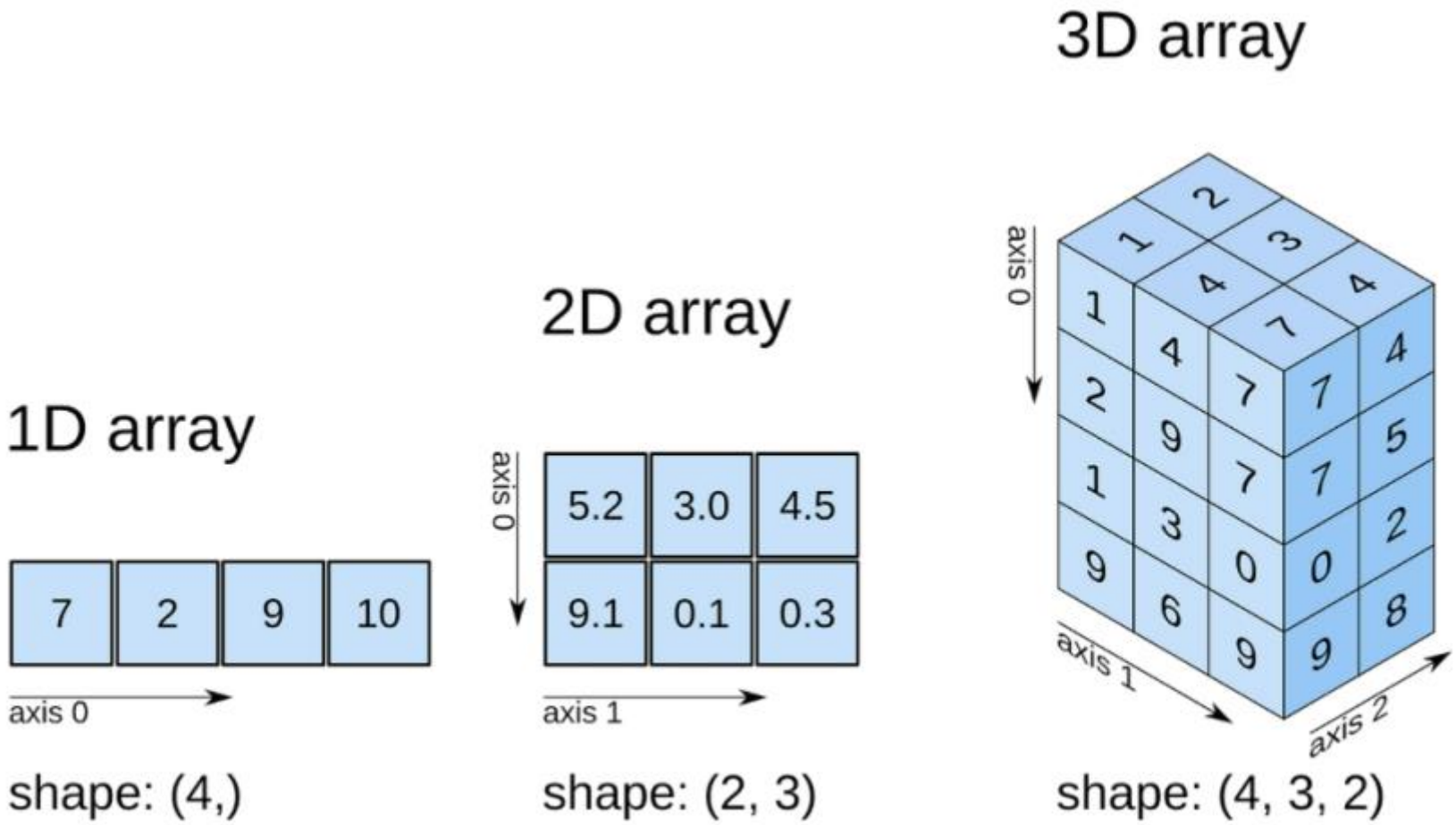
```
T = [1, 2, 8, 9, 13,14, 15]
max = T[0]
for i in range(1, len(T)):
    if T[i] > max :
        max = T[i]
print(max)
```

2D ARRAYS

DR. ADEL ABDELLI

01/03/2025

- The elements of a one-dimensional array are indexed consecutively (aligned cells). It is possible to arrange these cells in grids (two-dimensional arrays), cubes (three-dimensional arrays), and so on.



- **Showing elements of an Array:**

Let's say we have the following 2D array:

```
A = [ [25,  30,  28,  32],
      [15,  12,  18,  20],
      [10,  8,   12,  14],
      [8,   6,   9,   11]
    ]
```

➢ To show the 1st element (first row and first column) we write the following line:

```
print(A[0][0])
```

➢ To show the last element (last row and last column) we write one of the following line:

```
print(A[3][3])
```

```
print(A[-1][-1])
```

- **Showing elements of an Array:**

Let's say we have the following 2D array:

```
A = [ [25,  30,  28,  32],
      [15,  12,  18,  20],
      [10,  8,   12,  14],
      [8,   6,   9,   11]
    ]
```

➢ To show a complete row we write the following line:

```
print("showing the first row from the matrix", A[0])
```

➢ To show the last row we write one of the following line:

```
print("showing the last row from the matrix", A[3])
```

```
print("showing the last row from the matrix", A[-1])
```

- **Showing the shape of a 2D Array:**

Let's say we have the following 2D array (Matrix):

```
A = [ [25,  30,  28,  32],
      [15,  12,  18,  20],
      [10,  8,   12,  14],
      [8,   6,   9,   11]
    ]
```

➢ To show the number of rows we do the following:

```
print("number of rows is:", len(A))
```

➢ To show the number of columns we do the following:

```
print("number of columns is:", len(A[0])
```

- **Showing element of an Array:**

➢ To show all the element of the array one by one (from the first to the last) we do the following:

```
A = [ [25,  30,  28,  32],
      [15,  12,  18,  20],
      [10,  8,   12,  14],
      [8,   6,   9,   11]
    ]

for i in range( len( A ) ):
    for j in range( len ( A[i] ) ):
        print(A[i][j])
```

- ***Exercise 1:***
- We have the following matrix, and we need to write a Python code that search for the maximumm number:

```
A = [ [25,  30,  28],
      [15,  12,  18],
      [10,  8,   12],
      [8,   6,   9]
    ]
```

```
max = A[0][0]
for i in range(len(A)):
    for j in range(len(A[i])):
      if A[i][j] > max:
          max = A[i][j]
print(max)
```

- **Changing a value of an elements in a 2D Array:**

➢ To change the value of an element in 2D array we do the following:

Exemple 1:

```
A = [ [25,  30,  28,  32],
      [15,  12,  18,  20],
      [10,  8,   12,  14],
      [8,   6,   9,   11]
    ]

A[0][0] = 35
print(A)

#the output print A will be
[ [35,  30,  28,  32],
      [15,  12,  18,  20],
      [10,  8,   12,  14],
      [8,   6,   9,   11]
    ]
```

- **Adding a row to a 2D Array:**

➢ To add a row to an array we use the function append as following:

```
A = [ [25,  30,  28,  32],
      [15,  12,  18,  20],
      [10,  8,   12,  14],
      [8,   6,   9,   11]
    ]
new_row = [7, 8, 15, 2]
A.append(new_row)
#the content of A now will be:
[ [25,  30,  28,  32],
      [15,  12,  18,  20],
      [10,  8,   12,  14],
      [8,   6,   9,   11] ],
      [7,   8,   15,    2]
    ]
```

Note: The append function add elements at the end of the array,

- **Adding a row to a 2D Array:**

➢ Or by using insert function as following:

```
A = [ [25,  30,  28,  32],
      [15,  12,  18,  20],
      [10,  8,   12,  14],
      [8,   6,   9,   11]
    ]

new_row = [7, 8, 15, 2]
A.insert(1,new_row)
#the content of A now will be:
[ [25,  30,  28,  32],
      [7,   8,   15,    2],
      [15,  12,  18,  20],
      [10,  8,   12,  14],
      [8,   6,   9,   11] ]
]
```

- **Deleting a row from a 2D Array:**

➢ To delete a row we use the pop function as following:

```
A = [ [25,  30,  28,  32],
      [15,  12,  18,  20],
      [10,  8,   12,  14],
      [8,   6,   9,   11]
    ]
```

Here we delete the second row:

```
A.pop(1)

#the content of A now will be:
[ [25,  30,  28,  32],
      [10,  8,   12,  14],
      [8,   6,   9,   11] ]
]
```

- **Deleting an element from a 2D Array:**

➢ To delete an element from a 2D Array (Matrix) we use the pop function as following:

```
A = [ [25,  30,  28,  32],
      [15,  12,  18,  20],
      [10,  8,   12,  14],
      [8,   6,   9,   11]
    ]
```

Here we delete the last element from 2nd row:

```
A[1].pop(3)

#the content of A now will be:

[ [25,  30,  28,  32],
      [15,  12,  18],
      [10,  8,   12,  14],
      [8,   6,   9,   11]
    ]
```

Note: here the 2D array isn't anymore a matrix

- **Removing a Specific Element from a Row Using remove():**

➢ To delete a specific element from a 2D Array we use the remove function as following:

```
A = [ [25,  30,  28,  32],
      [15,  12,  18,  20],
      [10,  8,   12,  14],
      [8,   6,   9,   11]
    ]
```

Here we delete the element 12 from the 3rd row:

```
A[2].remove(12)

#the content of A now will be:
[ [25,  30,  28,  32],
      [15,  12,  18, 20],
      [10,  8,  14],
      [8,   6,   9,   11]
    ]
```

Note: here the 2D array isn't anymore a matrix

- **Removing a Column from All Rows:**

➢ To delete a column from a 2D Array we do the following:

```
A = [ [25,  30,  28,  32],
      [15,  12,  18,  20],
      [10,  8,   12,  14],
      [8,   6,   9,   11]
    ]
```

Here we delete the element 2nd column:

```
for i in range( len (A) ):

    A[i].pop(1)

#the content of A now will be:
A = [ [25,  28,  32],
      [15,  18,  20],
      [10,   12,  14],
      [8,   9,   11]
    ]
```

- **Filling a matrix (2D array) by asking the user:**

- To fill a 3x4 matrix we do the following:

```
A = []
n_row = 3
n_column = 4
for i in range(n_row):
    row = []
    print("filling the ",i+1," row")
    for j in range(n_column):
        a = int(input("enter a number"))
        row.append(a)
    A.append(row)
print(A)
```

THANK YOU FOR YOUR ATTENTION