Chapter 6

Array

Faculty of Engineering KMITL
King Mongkut's Institute of Technology Ladkrabang

Objective

- Understand the principle of Array.
- Write the program with Array structure.

Problem of variable

- In programming, one of the frequently arising problem is to handle numerous data of same type.

Write Program to get ID and Midterm Programming score of student room 1-10.

```
char    id0001[9],id0002[9],id0003[9],...,id1158[9],id1159[9];
float   point0001,point0002,point0003,...,point1158,point1159;
scanf   ("%s",id0001);
scanf   ("%f",&point0001);
...
scanf   ("%s",id1159);
scanf   ("%f",&point1159);
```

Use Array to fix the problem

```
#include<stdio.h>
#include<conio.h>
int main()
{
char id[1159][9]; float point[1159]; int i;
for (i=0;i<1159;i++)
{
scanf ("%s",id[i]);
scanf ("%f",&point[i]);
}
return 0;
}
```

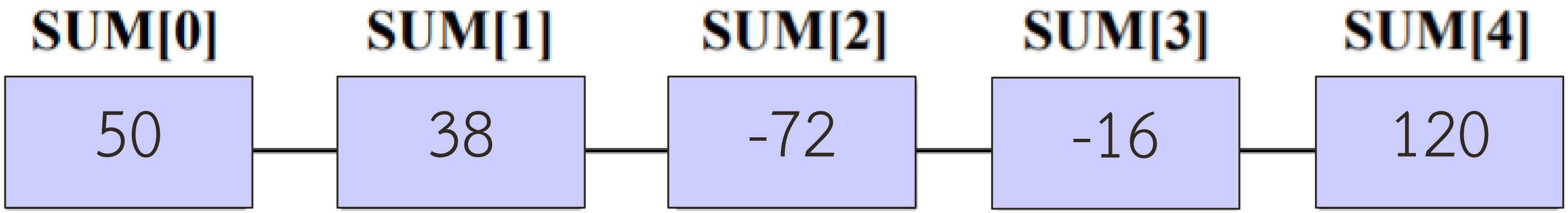
Reference..www.ce.kmitl.ac.th King Mongkut’s Institute of Technology Ladkrabang

Array

An **array** is a collection of data that holds fixed number of values of same type. The size and type of arrays cannot be changed after its declaration.

<u>Ex1</u>

Array variable name **SUM** for keep integer value size



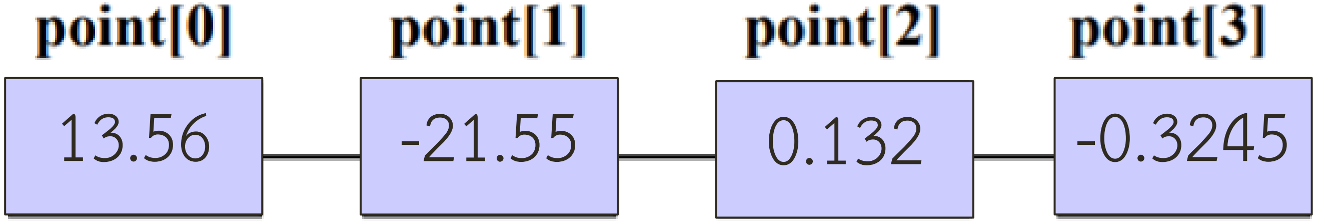
SUM[0]	SUM[1]	SUM[2]	SUM[3]	SUM[4]
50	38	-72	-16	120

Reference..www.ce.kmitl.ac.th King Mongkut's Institute of Technology Ladkrabang

Array

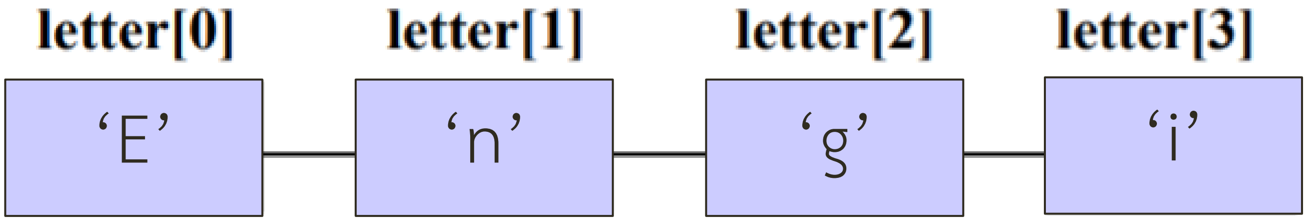
<u>Ex2</u>

Array variable name **point** for keep float value size 4



point[0]	point[1]	point[2]	point[3]
13.56	-21.55	0.132	-0.3245

Array variable name **letter** for keep char value size 4

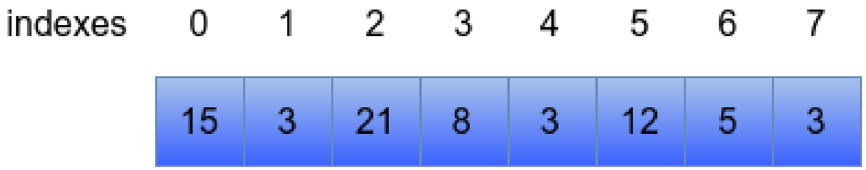


letter[0]	letter[1]	letter[2]	letter[3]
‘E’	‘n’	‘g’	‘i’

Reference..www.ce.kmitl.ac.th King Mongkut’s Institute of Technology Ladkrabang

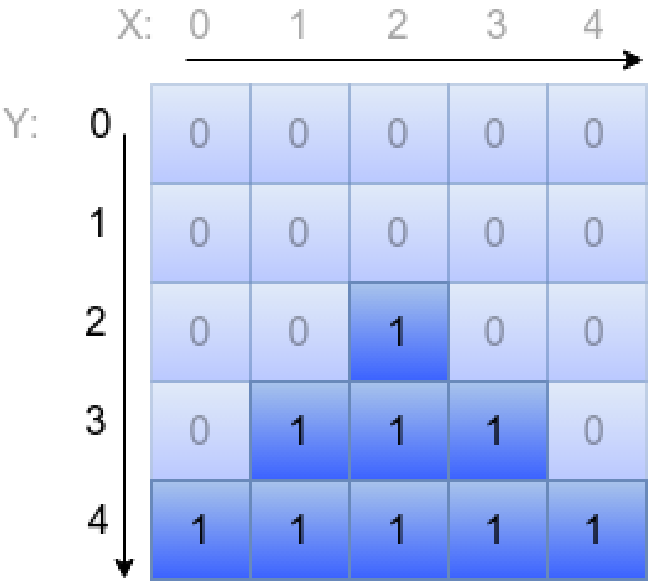
Array

One Dimensional Array



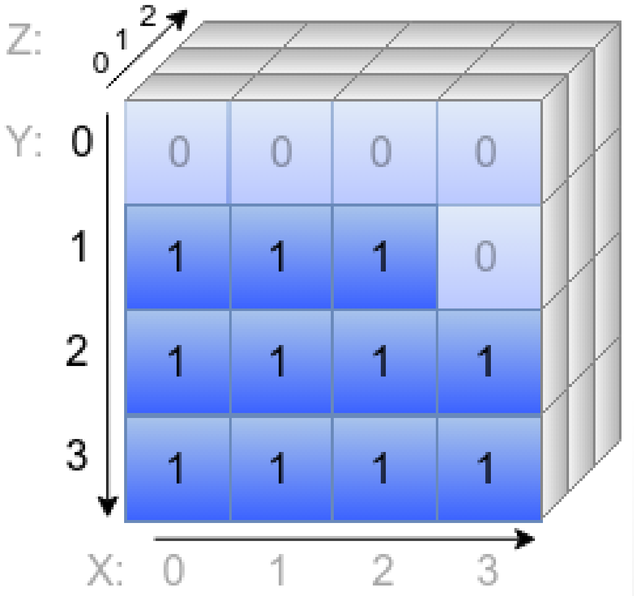
indexes	0	1	2	3	4	5	6	7
	15	3	21	8	3	12	5	3

Two Dimensional Array



Y: \ X:	0	1	2	3	4
0	0	0	0	0	0
1	0	0	0	0	0
2	0	0	1	0	0
3	0	1	1	1	0
4	1	1	1	1	1

Three Dimensional Array



Y: \ X:	0	1	2	3
0	0	0	0	0
1	1	1	1	0
2	1	1	1	1
3	1	1	1	1

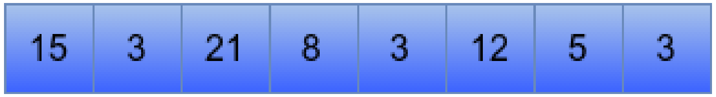
https://www.ict.social/csharp/basics/multidimensional-arrays-in-csharp-net

Array

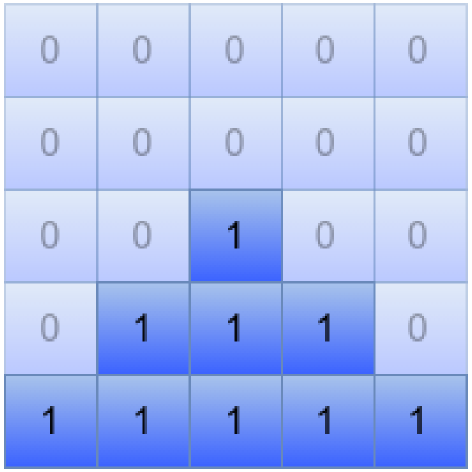
One Dimensional Array

Array that keep data in one row.

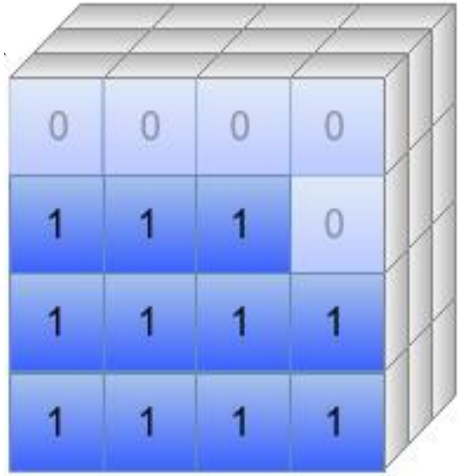
(use order to reference data)



15	3	21	8	3	12	5	3



0	0	0	0	0
0	0	0	0	0
0	0	1	0	0
0	1	1	1	0
1	1	1	1	1



0	0	0	0
1	1	1	0
1	1	1	1
1	1	1	1

Two Dimensional Array

Array that keep data by use row and column to reference data.

Three Dimensional Array

Array that keep data in cube shape by use row, column and deep to reference data.

https://www.ict.social/csharp/basics/multidimensional-arrays-in-csharp-net

Array 1 Dimensional

```
type        array-name [n] ;
```

type is type of variable that will be created

- int	integer variable
- float	decimal variable
- char	character variable

array-name is name of array variable

n is size of array variable

Array 1 Dimensional

Defined value into array

```
type array-name [n] = {value-1, value-2,…,value-n};
```

value-1, value-2,…,value-n is value define into array. The value must be the same type which defined at the front.

Reference..www.ce.kmitl.ac.th King Mongkut's Institute of Technology Ladkrabang

Array 1 Dimensional

```
#include<stdio.h>
#include<conio.h>
int main()
{
    int number[3] = {23, -186, 43};
    float value_2[5]={0.98,43.213,-3.47,52.08,-0.987};
    char vowel[5] = {'a','e','i','o','u’};
    char name[9] = {'E','n','g','i','n','e','e','r','\0’};
    return 0;
}
```

Reference..www.ce.kmitl.ac.th King Mongkut’s Institute of Technology Ladkrabang

Reference Array

```
int    year[5] = {2001,2542,1999,2000,2521};
```

	year[0]	year[1]	year[2]	year[3]	year[4]
year[5]	**2001**	**2542**	**1999**	**2000**	**2521**

```
printf ("%d\n",year[0]);

printf ("%d\n",year[1]);

printf ("%d\n",year[2]);

printf ("%d\n",year[3]);

printf ("%d\n",year[4]);
```

```
2001
2542
1999
2000
2521
```

Reference Array

```
#include<stdio.h>
#include<conio.h>
int main()
{
int year[5] = {2001,2542,1999,2000,2521};
printf ("%d\n",year[2]);
printf ("%d\n\n",year[4]);
year[0] = 2545;
printf ("%d\n",year[0]);
return 0;
}
```

```
1999
2521

2545
```

	year[0]	year[1]	year[2]	year[3]	year[4]
year[5]	2545	2542	1999	2000	2521

Example : Program analyze height of n people

Write the Flowchart and Program to get number of student then get height of n people and analyze range of height from this table and show average.

0 - 160	**161 – 170**
171 - 180	**181 – 200**

Example : Program analyze height of n people

- Output Analysis

    - Number of student in each range

    - Height average of all student

    - Height of all student

- Input Analysis

    - Number of all student and height of each student

Example : Program analyze height of n people

- Process Analysis

  - Program get number of student

  - Loop program get height of all student

  - Loop program check and sum height of student

  - Calculate average

Example : Program analyze height of n people

- Variable Define

num is integer to keep number of student

count is integer to check variable position and count

range1=0, range2=0, range3=0, range4=0

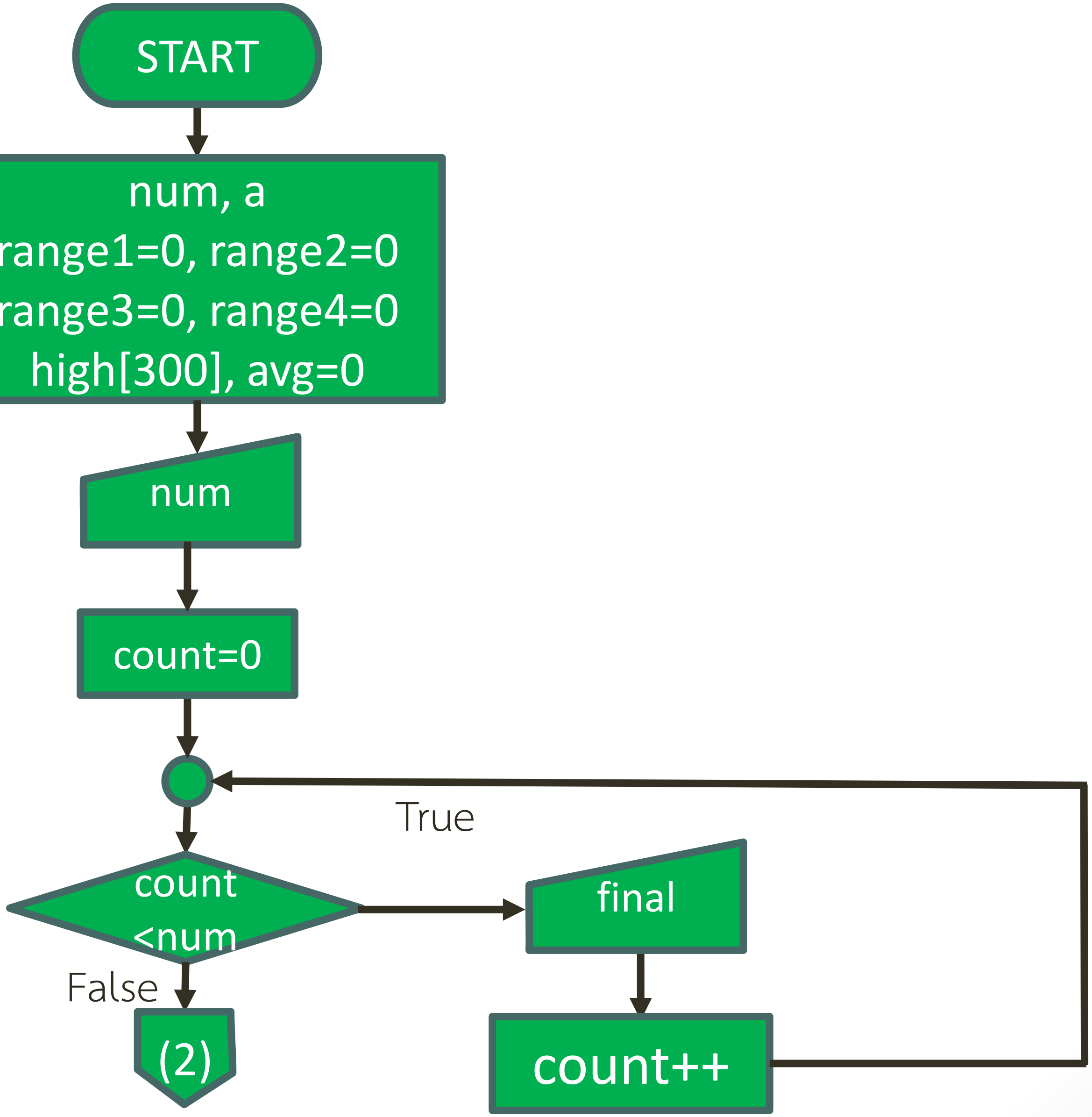
is integer to keep number of student in each range

high[300] is float array to keep height

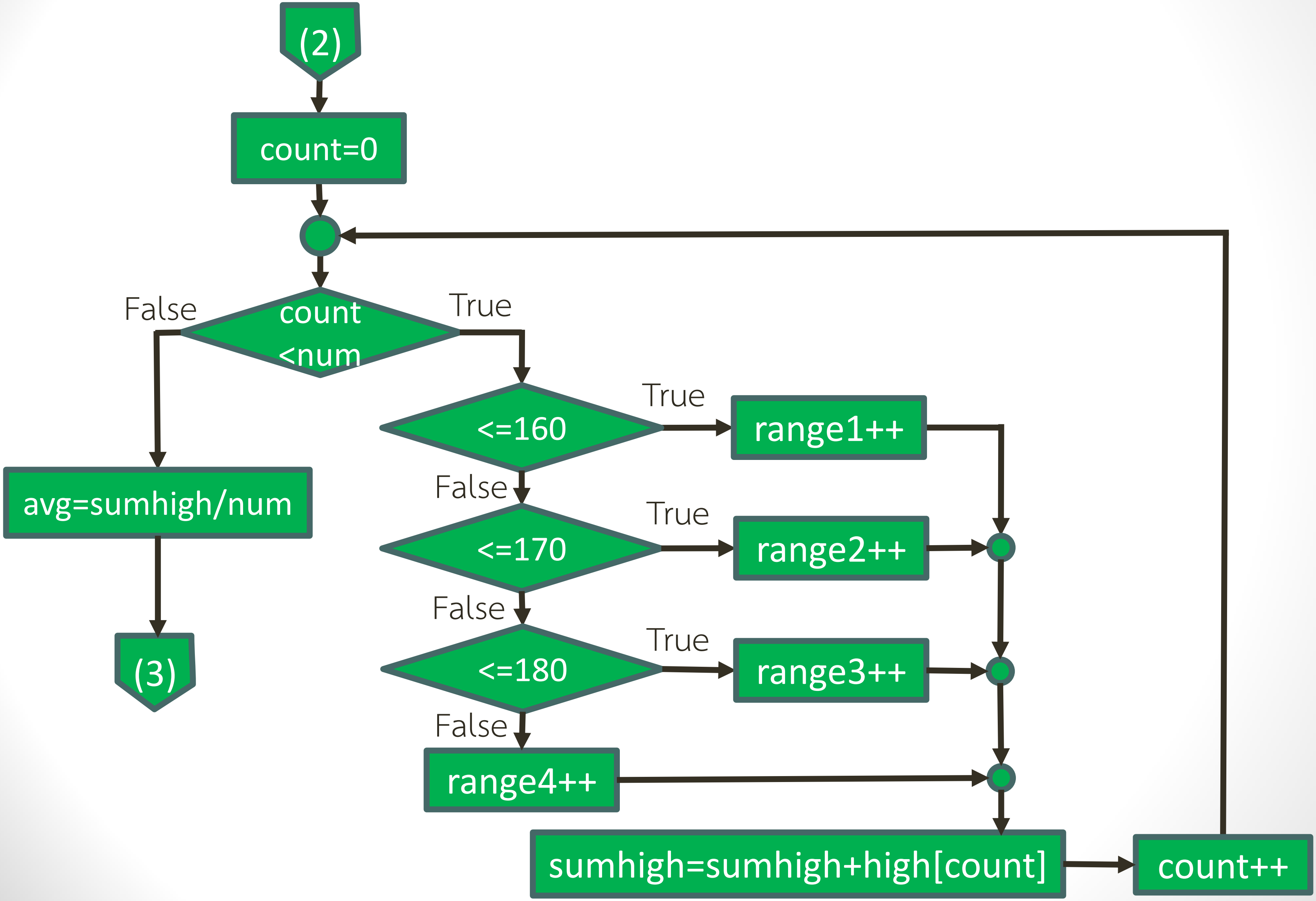
sumhigh is float to keep summation of height

avg is float to keep average of all student

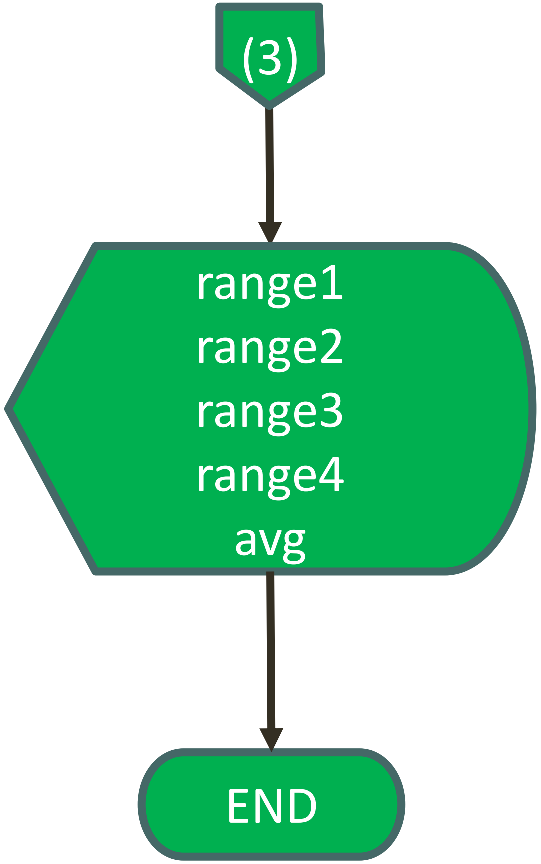
Example : Program analyze height of n people



Example : Program analyze height of n people



Example : Program analyze height of n people



Example : Program analyze height of n people

```
#include<stdio.h>
#include<conio.h>
int main()
{
int       num,count,range1=0,range2=0,range3=0,range4=0;
float     high[300],sumhigh=0,avg=0;
printf    ("Please enter number of student : ");
scanf     ("%d",&num);
for       (count=0; count<num; count++)
{
    printf ("Student %2d : ",count+1);
    scanf ("%f",&high[count]);
}
```

Reference..www.ce.kmitl.ac.th King Mongkut’s Institute of Technology Ladkrabang

Example : Program analyze height of n people

```
for (count=0; count<num; count++)
{
    if (high[count]<=160)
        range1++;
    else if (high[count]<=170)
        range2++;
    else if (high[count]<=180)
        range3++;
    else
        range4++;
    sumhigh = sumhigh + high[count];
}
avg = sumhigh/num;
```

Reference..www.ce.kmitl.ac.th King Mongkut's Institute of Technology Ladkrabang

Example : Program analyze height of n people

```
    printf ("\n 0 - 160 : %3d",range1);

    printf ("\n161 - 170 : %3d",range2);

    printf ("\n171 - 180 : %3d",range3);

    printf ("\n181 - 200 : %3d",range4);

    printf ("\n\nAverage : %f ",avg);

    return 0;

}
```

String Array

```
char  subject[11] = {“C language”};
```

or

```
char subject[11] = {‘C’, ‘ ’, ‘l’, ‘a’, ‘n’, ‘g’, ‘u’, ‘a’, ‘g’, ‘e’, ‘\0’};
```

	[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]	[10]
subject	C		l	a	n	g	u	a	g	e	\0

```
char    name[9] = {“Engineer”};
```

	[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]
name	E	n	g	i	n	e	e	r	\0

Reference..www.ce.kmitl.ac.th King Mongkut’s Institute of Technology Ladkrabang

Wrong Example String Array

```
#include<stdio.h>
#include<conio.h>
int main()
{
char      sentence [22] = “Welcome to my country”;
char      word [9] = {‘T’, ‘h’, ‘a’, ‘i’, ‘l’, ‘a’, ‘n’, ‘d’, ‘\0’};
char      not_word [4] = {‘l’, ‘o’, ‘v’, ‘e’};
printf    (“Message1 = %s\n“, sentence);
printf    (“Message2 = %s\n“, word);
printf    (“Message3 = %s\n“, not_word);
return 0;
}
```

```
Message1 = Welcome to my country
Message2 = Thailand
Message3 = loveThailand
```

Array 2 Dimensional

```
type        array-name [n] [m] ;
```

type is type of variable that will be created

- int integer variable

- float decimal variable

- char character variable

array-name is name of array variable

n is row of array variable

m is column of array variable

Reference..www.ce.kmitl.ac.th King Mongkut's Institute of Technology Ladkrabang

Array 2 Dimensional

Defined value into array

```
type array-name [n] [m]= {value-1-1, value-1-2,…,value-1-m,
                                   value-2-1, value-2-2,…,value-2-m,
                                   …,
                                    value-n-1, value-n-2,…,value-n-m};
```

value-1-1, value-1-2,…,value-1-n,…,value-n-m

is value that define into array. The value must be the same type which defined at the front.

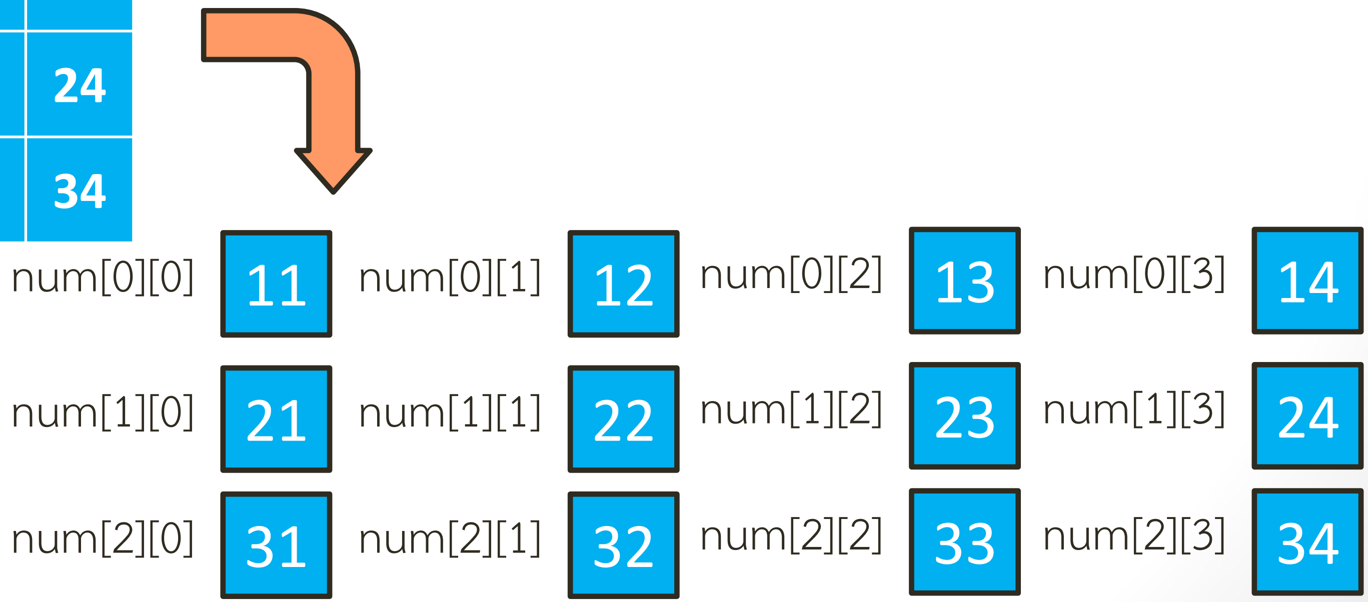
Reference..www.ce.kmitl.ac.th King Mongkut's Institute of Technology Ladkrabang

Define value to Array 2 dimensional

```
int      num[3][4] = {11, 12, 13, 14,
                      21, 22, 23, 24
                      31, 32, 33, 34};

int      num[3][4] = {11, 12, 13, 14, 21, 22, 23, 24, 31, 32, 33, 34};
```

11	12	13	14
21	22	23	24
31	32	33	34



num[0][0] 11 num[0][1] 12 num[0][2] 13 num[0][3] 14

num[1][0] 21 num[1][1] 22 num[1][2] 23 num[1][3] 24

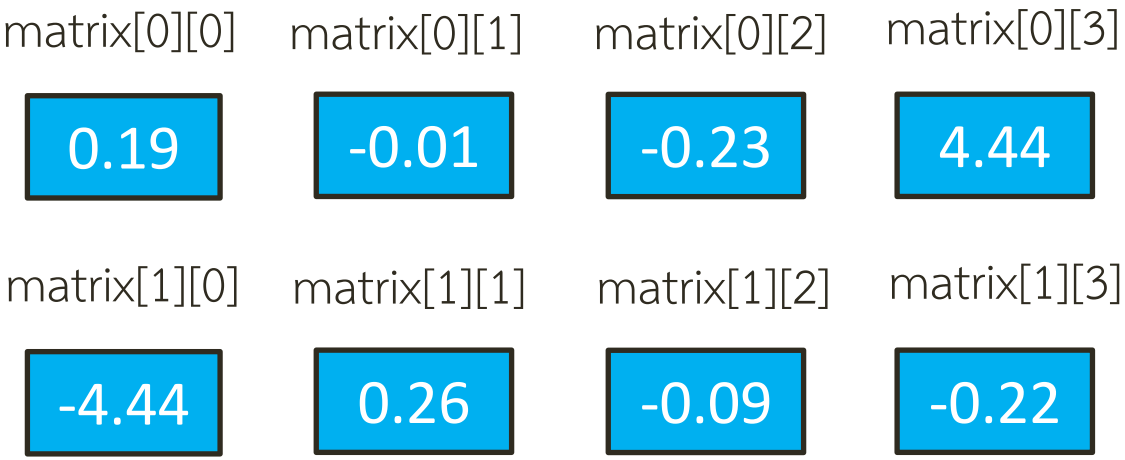
num[2][0] 31 num[2][1] 32 num[2][2] 33 num[2][3] 34

Reference..www.ce.kmitl.ac.th King Mongkut's Institute of Technology Ladkrabang

Define value to Array 2 dimensional

```
float    matrix[2][4] = {0.19, -0.01, -0.23, 4.44,
                         -4.44, 0.26, -0.09, -0.22};

float    matrix[2][4] = {0.19, -0.01, -0.23, 4.44, -4.44, 0.26, -0.09, -0.22};
```



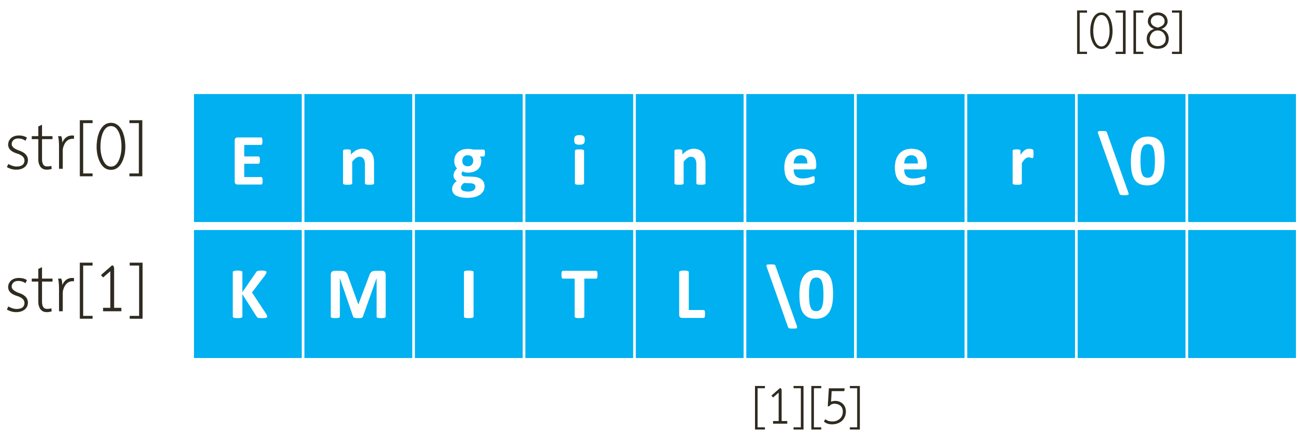
matrix[0][0]	matrix[0][1]	matrix[0][2]	matrix[0][3]
0.19	-0.01	-0.23	4.44
matrix[1][0]	matrix[1][1]	matrix[1][2]	matrix[1][3]
-4.44	0.26	-0.09	-0.22

Reference..www.ce.kmitl.ac.th King Mongkut's Institute of Technology Ladkrabang

Define value to Array 2 dimensional

```
char   str[2][10] = {“Engineer”,
                     “KMITL”};

char   str[2][10] = {“Engineer”, “KMITL”};
```



[0][8]

str[0]	E	n	g	i	n	e	e	r	\0	
str[1]	K	M	I	T	L	\0				

[1][5]

Example : Program get and show Matrix data

Write the Program to get integer number in Matrix data format then shows in the form of 3x3 Matrix

```
Enter numbers [0] [0] : 1
Enter numbers [0] [1] : 2
Enter numbers [0] [2] : 3
Enter numbers [1] [0] : 4
Enter numbers [1] [1] : 5
Enter numbers [1] [2] : 6
Enter numbers [2] [0] : 7
Enter numbers [2] [1] : 8
Enter numbers [2] [2] : 9
```

```
***Matrix***
   1    2    3
   4    5    6
   7    8    9
```

Example : Program get and show Matrix data

```
#include<stdio.h>
#include<conio.h>
int main()
{
    int      matrix [3] [3] ,r ,c;
    for (r=0; r<3; r++)
    {
          for(c=0; c<3; c++)
          {
          printf      (“Enter numbers [%d] [%d] : “,r,c);
          scanf       (“%d”, &matrix [r] [c]);
    }
}
```

Example : Program get and show Matrix data

```
    printf          (“\n*** Matrix ***\n”);
    for (r=0; r<3; r++)
    {
        for(c=0; c<3; c++)
        {
            printf (“%5d”, matrix [r] [c]);
        }
        printf (“\n”);
    }
    return 0;
}
```

Example : Program calculate Matrix summation

From Defined Array variable. Write the program to calculate summation of value in each row and each column by keep summation data in to variable row[ ], column[ ]

```
int num[3][4] = {1, 2, 3, 4,
                 2, 3, 4, 5,
                 3, 4, 5, 6};
```

Example : Program calculate Matrix summation

```
 ***Show Matrix***

    1       2       3       4

    2       3       4       5

    3       4       5       6


Sum of row [0] = 10

Sum of row [1] = 14

Sum of row [2] = 18

Sum of column [0] = 6

Sum of column [1] = 9

Sum of column [2] = 12

Sum of column [3] = 15
```

Example : Program calculate Matrix summation

```
#include<stdio.h>
#include<conio.h>
int main()
{
        int num[3][4] = { 1, 2, 3, 4,
                                      2, 3, 4, 5,
                                      3, 4, 5, 6 };
        int r,c,row[3]={0,0,0},column[4]={0,0,0,0};
        /* Display Matrix */
        printf ("\n*** Show Matrix ***\n\n");
        for (r=0; r<3; r++)
        {
             for(c=0; c<4; c++)
                printf ("%5d ",num[r][c]);
             printf ("\n\n");
}
```

Example : Program calculate Matrix summation

```
    /* Summation Matric */
    for (r=0; r<3; r++)
        for(c=0; c<4; c++)
        {
            row[r] = row[r] + num[r][c];
            column[c] = column[c] + num[r][c];
        }
    /* Display Summation */
    printf ("\n\n");
    for (r=0; r<3; r++)
        printf ("sum of row [%d] = %d\n",r,row[r]);
    for (c=0; c<4; c++)
        printf ("sum of column [%d] = %d\n",c,column[c]);
    return 0;
}
```

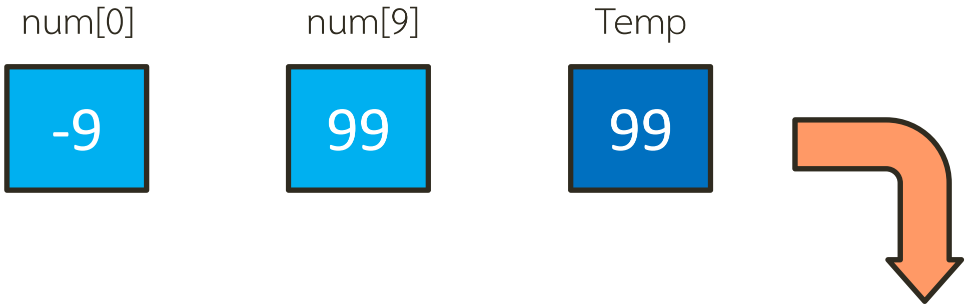
Swap Array value

	num[0]	num[1]	num[2]	num[3]	num[4]	num[5]	num[6]	num[7]	num[8]	num[9]
num[10]	99	19	1	23	0	15	10	23	-26	-9

```
temp = num[0];

num[0] = num[9];

num[9] = temp;
```



num[0]	num[9]	Temp
-9	99	99

	num[0]	num[1]	num[2]	num[3]	num[4]	num[5]	num[6]	num[7]	num[8]	num[9]
num[10]	-9	19	1	23	0	15	10	23	-26	99

Reference..www.ce.kmitl.ac.th King Mongkut's Institute of Technology Ladkrabang

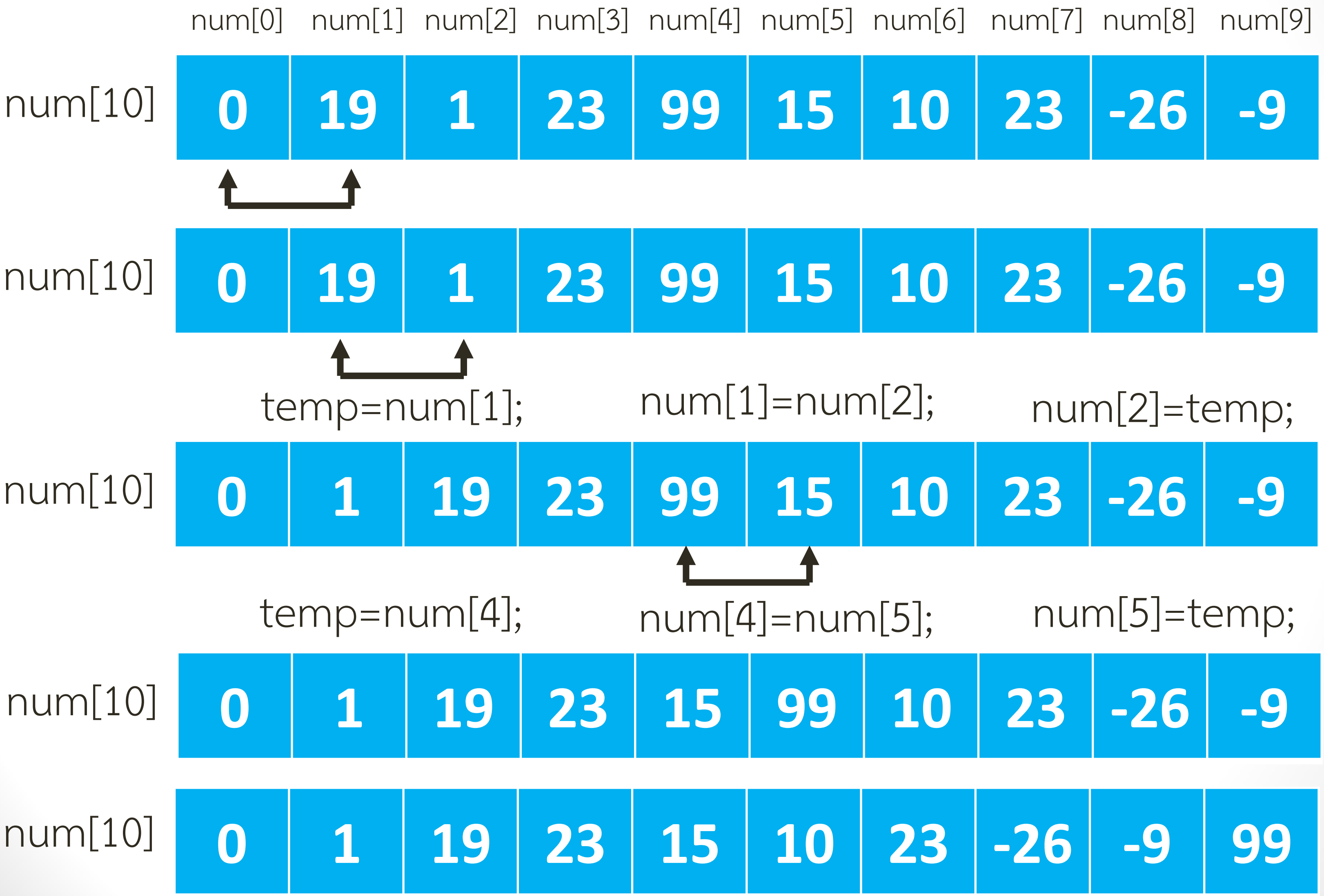
Example : Find the max value

Write the program to get integer 10 value.

To find max value.

Move the max value to last Array position.

Example : Find the max value



	num[0]	num[1]	num[2]	num[3]	num[4]	num[5]	num[6]	num[7]	num[8]	num[9]
num[10]	0	19	1	23	99	15	10	23	-26	-9
num[10]	0	19	1	23	99	15	10	23	-26	-9

temp=num[1]; num[1]=num[2]; num[2]=temp;

num[10]	0	1	19	23	99	15	10	23	-26	-9

temp=num[4]; num[4]=num[5]; num[5]=temp;

num[10]	0	1	19	23	15	99	10	23	-26	-9
num[10]	0	1	19	23	15	10	23	-26	-9	99

Reference..www.ce.kmitl.ac.th King Mongkut’s Institute of Technology Ladkrabang

Example : Find the max value

```
#include<stdio.h>
#include<conio.h>
#define SIZE 10
int main()
{
      int num[SIZE],temp,n;
      for (n=0; n<SIZE; n++)
      {
         printf ("Enter num[%d] : ",n+1);
         scanf ("%d",&num[n]);
      }
```

Example : Find the max value

```
    for (n=0; n<SIZE-1; n++)
    {
        if (num[n]>num[n+1])
        {
            temp = num[n+1];
            num[n+1] = num[n];
            num[n] = temp;
        }
    }
    printf ("The maximum number = %d",num[SIZE-1]);
    return 0;
}
```

Exercise

1.Define Matrix A size 3x3 is

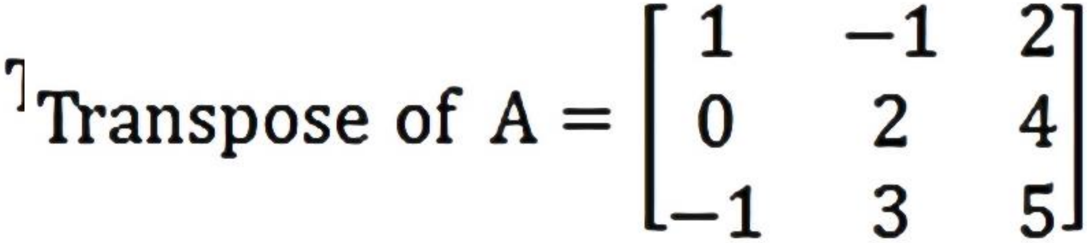
$$A = \begin{bmatrix} 1 & 0 & -1 \\ -1 & 2 & 3 \\ 2 & 4 & 5 \end{bmatrix}$$

Write the program to find Diagonal matrix of A by put the result into Array A

$$\textbf{Diagonal matrix of } A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 5 \end{bmatrix}$$

Exercise

2.From Exercise1 Write the program to find Transpose of Matrix A by put the result into Matrix A



$$\text{Transpose of } A = \begin{bmatrix} 1 & -1 & 2 \\ 0 & 2 & 4 \\ -1 & 3 & 5 \end{bmatrix}$$

Exercise

2. From Matrix A in Exercise1 Write the program to calculate the result of AxA

$$\mathbf{A}\times\mathbf{A}=\begin{bmatrix}(1)(1)+(0)(-1)+(-1)(2) & (1)(0)+(0)(2)+(-1)(4) & (1)(-1)+(0)(3)+(-1)(5)\\(-1)(1)+(2)(-1)+(3)(2) & (-1)(0)+(2)(2)+(3)(4) & (-1)(1)+(2)(3)+(3)(5)\\(2)(1)+(4)(-1)+(5)(2) & (2)(0)+(4)(2)+(5)(4) & (2)(-1)+(4)(3)+(5)(5)\end{bmatrix}$$

$$\mathbf{A}\times\mathbf{A}=\begin{array}{|ccc|}\hline -1 & -4 & -6\\ 3 & 16 & 22\\ 8 & 28 & 35\\ \hline\end{array}$$