Question 01:

Write a C++ program to request the user to enter a number and print a pattern as follows:

```
Sample Output:
Enter number: 3
Output:
1
3 3
5 5 5
```

Question 02:

Write a C++ to read a number from the user and check whether that number is **niven number**

or not.

Hint: A Niven number is that a number which is divisible by its sum of digits.

Refer to an example given below.

```
Input: 126
sum of digits:1+2+6 = 9
126  is divisible by 9
Hence 126 is a niven number.
```

```
Sample Output:
Enter a number: 126
The number is a niven number.
```

Question 03:

"Pastry Kade" is a famous pastry shop and they are promoting their new arrivals to the customers. Following table shows the information about the new food items.

Food Item	Description	Discount	Price
AK012	Cheese Pizza	2%	120
AK013	Pasta	3%	350
AK014	Wet fried rice	3%	350
AK015	Vegetable salad with cone pudding	5%	250

If the user buys a food item listed above, they will add the discount mentioned.

a) Write a program to allow the user to enter the food item and the number of items needed. Find the total price to be paid for the selection. **You should use nested selection statements.**

b) Modify your program to display an error message "Invalid Food Item", if the user inputs a wrong Food Item which is not available in the list.

c) Modify the program to handle many inputs. After displaying the total price, the program should display a prompt "Do you have more data?". If the user inputs "y" or "Y", program should ask for the next food item. If the user inputs "n" or "N", program should terminate and display the total cost.

Note: format the output into two decimal places.

```
Sample Output:
Enter the food item: AKO12
Enter the number of items needed: 2
Do You have more data? Y
Enter the food item: AKO13
Enter the number of items needed: 1
Do You have more data? N
Total Price is: 574.70
```

Question 04:

a) Create a structure called **Employee** that includes employee ID (string), name of the employee (string), over-time fee (float) and the number of over-time hours during the weekdays (int array).

b) Write a function called **getEmp( )** which is the data type of Employee that reads the details of Employee and store them in the variable of the Employee structure.

```
Employee getEmp(Employee e);
```

c) Write a function called **calOTpayment( )** which takes three parameters, over-time fee of the employee, number of over-time hours in weekdays (5 days) array and the size of the array. Find the total payment for the employee and print the total Over-time fee for Weekdays.

d) Call the getEmp( ) and calOTpayment( ) functions in the main function to print the following output as required.

e) In the main function change the Over time Fee of the Employee into 600.00 using pointers dynamically and print the new Overtime Fee to the screen.

```
Sample Output:
Enter Employee ID: MKT12345
Enter the Name of the Employee: Kamal
Enter the Over-time Fee: 450.00
Enter the number of OT Hours for Day 1:1
Enter the number of OT Hours for Day 2:2
Enter the number of OT Hours for Day 3:0
Enter the number of OT Hours for Day 4:1
Enter the number of OT Hours for Day 5:2

Total Over Time Payment: 2700
The new overtime fee is: 600
```

Question 05:

In a school 5 students were selected for a volleyball team. The heights were recorded in an array to do an analysis about the student heights.

Student	1	2	3	4	5
Array *index*	0	1	2	3	4
Height	150	180	165.5	175	165.6

a) Write a function called **inputHeights ()** which will take float array and the integer size of the array as parameters and fill the array with keyboard inputs.

b) Write a function called **findMaxHeight ()** which will take float array and the integer size of the array as parameters to find and return the maximum height in the array.

c) Write a function called **findMinHeight ()** which will take float array and the integer size of the array as parameters to find and return the minimum height in the array.

d) Write a function called **findTotalHeight ()** to find and return the total of the heights of the students. The function should take the will take float array and the integer size of the array as parameters.

e) Write a function called **printHeights ()** to print the height of all the students selected and the function should take the will take float array and the integer size of the array as parameters.

Note: The function should print the Output as follows:

```
The Height of the Student 1: 150
The Height of the Student 2: 180
--------------------
```

f) Implement the main function to do the followings:

- Create a float array with the name **Height**. The arrays are of size 5.

- Insert Height of each student to Height array using **InputHeights ()** function.

- Find and print the maximum height of the set of 5 students using the function **findMaxHeight ()**.

- Find and print the minimum height of the set of 5 students using the function **findMinHeight ()**.

- Find and print the total heights of the students in the team using **findTotalHeight ()** function.

- Print the set of heights using **printHeights ()** function.