

**Definition:** Flowcharts are nothing but the graphical representation of the data or the algorithm for a better understanding of the code visually. It displays step-by-step solutions to a problem, algorithm, or process.

फ़्लोचार्ट और कुछ नहीं बल्कि डेटा या एल्गोरिथम का ग्राफिकल प्रतिनिधित्व(representation) है जो कोड को बेहतर ढंग से समझने के लिए है। यह किसी समस्या, एल्गोरिथम या प्रक्रिया के चरण-दर-चरण समाधान प्रदर्शित करता है।

## Use of a flowchart

Following are the uses of a flowchart:

- It is a pictorial representation of an algorithm that increases the readability of the program.
- Complex programs can be drawn in a simple way using a flowchart.
- It helps team members get an insight into the process and use this knowledge to collect data, detect problems, develop software, etc.
- A flowchart is a basic step for designing a new process or add extra features.
- Communication with other people becomes easy by drawing flowcharts and sharing them.

यह एक एल्गोरिथम का सचित्र प्रतिनिधित्व है जो प्रोग्राम की पठनीयता को बढ़ाता है।

- फ़्लोचार्ट का उपयोग करके जटिल कार्यक्रमों को सरल तरीके से तैयार किया जा सकता है।

- यह टीम के सदस्यों को प्रक्रिया में एक अंतर्दृष्टि प्राप्त करने और डेटा एकत्र करने, समस्याओं का पता लगाने, सॉफ्टवेयर विकसित करने आदि के लिए इस ज्ञान का उपयोग करने में मदद करता है।

- एक फ़्लोचार्ट एक नई प्रक्रिया को डिज़ाइन करने या अतिरिक्त सुविधाओं को जोड़ने के लिए एक बुनियादी कदम है।

- फ़्लोचार्ट बनाकर और उन्हें साझा करने से अन्य लोगों के साथ संचार आसान हो जाता है।

## When to use flowchart

Flowcharts are mainly used in the below scenarios:

- It is most importantly used when the programmers make projects. As a flowchart is a basic step to make the design of projects pictorially, it is preferred by many.
- When the flowcharts of a process are drawn, the programmer understands the non-useful parts of the process. So flowcharts are used to separate useful logic from the unwanted parts.
- Since the rules and procedure of drawing a flowchart are universal, flowchart serves as a communication channel to the people who are working on the same project for better understanding.
- Optimizing a process becomes easier with flowcharts. The efficiency of code is improved with the flowchart drawing.

यह सबसे महत्वपूर्ण रूप से उपयोग किया जाता है जब प्रोग्रामर प्रोजेक्ट बनाते हैं। चूंकि फ़्लोचार्ट परियोजनाओं के डिज़ाइन को सचित्र रूप से बनाने के लिए एक बुनियादी कदम है, इसलिए इसे कई लोगों द्वारा पसंद किया जाता है।

- जब किसी प्रक्रिया का फ़्लोचार्ट तैयार किया जाता है, तो प्रोग्रामर प्रक्रिया के अनुपयोगी भागों को समझता है। इसलिए फ़्लोचार्ट का उपयोग उपयोगी तर्क को अवांछित भागों से अलग करने के लिए किया जाता है।

- चूंकि फ़्लोचार्ट बनाने के नियम और प्रक्रिया सार्वभौमिक हैं, इसलिए फ़्लोचार्ट उन लोगों के लिए संचार चैनल के रूप में कार्य करता है जो बेहतर समझ के लिए एक ही प्रोजेक्ट पर काम कर रहे हैं।

- फ़्लोचार्ट के साथ किसी प्रक्रिया को अनुकूलित करना आसान हो जाता है। फ़्लोचार्ट ड्राइंग के साथ कोड की दक्षता में सुधार होता है।

## Advantages of Flowchart

- It is the most efficient way of communicating the logic of system.
- It act like a guide for blueprint during program designed.
- It also helps in debugging process.
- Using flowchart we can easily analyze the programs.
- flowcharts are good for documentation.

यह प्रणाली के तर्क को संप्रेषित करने का सबसे कुशल तरीका है।






- यह डिजाइन किए गए कार्यक्रम के दौरान ब्लूप्रिंट के लिए एक गाइड की तरह काम करता है।
  - यह डिबगिंग प्रक्रिया में भी मदद करता है।
  - फ्लोचार्ट का उपयोग करके हम आसानी से कार्यक्रमों का विश्लेषण कर सकते हैं।
- फ्लोचार्ट दस्तावेज़ीकरण के लिए अच्छे हैं।

### **Disadvantages of Flowchart**

- Flowcharts are difficult to draw for large and complex programs.
- It does not contain the proper amount of details.
- Flowcharts are very difficult to reproduce.
- Flowcharts are very difficult to modify.

बड़े और जटिल कार्यक्रमों के लिए फ्लोचार्ट बनाना कठिन होता है।

- इसमें उचित मात्रा में विवरण नहीं है।
- फ्लोचार्ट को पुनः प्रस्तुत करना बहुत कठिन है।
- फ्लोचार्ट को संशोधित करना बहुत कठिन है।

Symbol	Name	Function
	Start/end	An oval represents a start or end point
	Arrows	A line is a connector that shows relationships between the representative shapes
	Input/Output	A parallelogram represents input or output
	Process	A rectangle represents a process
	Decision	A diamond indicates a decision

Draw a flowchart to find the greatest number among the 2 numbers.

**Find the sum of two numbers N and M**