

১৯তম শিক্ষক নিবন্ধন (বিষয় ভিত্তিক)

NTRCA ICT

Lecturer – Code(452)

মো. শাহরিয়ার নাজিম জয়

১৯তম শিক্ষক নিবন্ধন (বিষয় ভিত্তিক)

# NTRCA ICT

Lecturer- ICT (Code-452)

সম্পূর্ণ সিলেবাস অনুসারে রচিত ও সর্বোচ্চ MCQ সম্বলিত

## বইটির বৈশিষ্ট্য

- ☆ সহজভাষায় ব্যাখ্যা-বিশ্লেষণ
- ☆ টপিক অনুযায়ী MCQ
- ☆ অতিরিক্ত গুরুত্বপূর্ণ টপিক সংযুক্ত
- ☆ পর্যাপ্ত MCQ
- ☆ হাই ভোল্টেজ MCQ
- ☆ Self Test
- ☆ Model Test
- ☆ ৮৫ থেকে ৯০% কমন উপযোগী

Md. Shahriar Nazim Joy



SYNCHRONOUSBOI  
PUBLICATION

NTRCA ICT

Md. Shahriar Nazim Joy

সূচিপত্র

নং	অধ্যায়	বিষয়	পেইজ নং
1	Chapter-1: Structured and Object Oriented Programming (OOP) Concept	প্রোগ্রাম ডিজাইন মডেলঃ	1
		প্রোগ্রামের ধারণা	2
		অনুবাদক প্রোগ্রাম	6
		প্রোগ্রাম সংগঠন	6
		ডিবাগিং, অ্যালগোরিদম, ফ্লোচার্ট	7
		সি' প্রোগ্রামিং	10
		ডেটা টাইপ	12
		চলক, অপারেটর	16
		ইনপুট ও আউটপুট স্টেটমেন্ট	18
		কন্ট্রোল স্টেটমেন্ট	19
		অ্যারে (Array)	23
		ফাংশন (Function)	24
		কলিং ফাংশন (Calling Function)	25
		রিকার্সিভ ফাংশন	28
		হেডার ফাইল (Header File)	33
		টোকেন (Token)	34
		কনস্ট্যান্ট (Constant)	35
		পয়েন্টার (Pointer)	36
		IMPORTANT ICT	43
		SELF TEST	51
2	Chapter-2: Introduction to Software Engineering	সফটওয়্যার ইঞ্জিনিয়ারিং, ইতিহাস, SDLC	53
		প্রোগ্রামিং ভাষা	54
		Software এর প্রকৃতি	59
		Software development model: waterfall	59
		Software development model: agile	60
		Software development model: spiral	61
		Software development model: RDD	64
		Software development model: V model	64
		COCOMO মডেল	65
		Capability Maturity Model (CMM)	66
		Software engineering principles	68
		Specification and Verification	68
		Modeling and Design	69
		Software Project Management	71
		ইনফরমেশন সিস্টেম	71

		Business Information System	72
		সিস্টেম সফটওয়্যার (System Software)	73
		এপ্লিকেশন সফটওয়্যার (Application Software)	73
		স্ট্র্যাটেজিক ইনফরমেশন সিস্টেম	74
		ইনফরমেশন সিস্টেমের ডাইমেনশন	78
		ম্যানেজমেন্ট ইনফরমেশন সিস্টেম - MIS	79
		ডিসিশন সাপোর্ট সিস্টেম - DSS	79
		ফর্মওয়্যার, হিউম্যানওয়্যার, আইডিই	83
		Umbrella Activities	83
		রিকোয়ারমেন্ট ইঞ্জিনিয়ারিং	84
		ইউজার ইন্টারফেস, ডেটা ফ্লো ডায়াগ্রাম	85
		সফটওয়্যার ইঞ্জিনিয়ারিংয়ে টেস্টিং	88
		মডুলারাইজেশন (Modularization)	90
		SELF TEST	93
3	Chapter-3: Data Structure and Algorithm & Combinatorial Optimization	Data Structure, ডেটা স্ট্রাকচারের অপারেশন	95
		Time Space Tradeoff	97
		Searching Techniques- Linear and Binary Searching	99
		সর্টিং (Sorting)	100
		রিকার্সন (Recursion)	101
		ইনসার্সন সর্ট (Insertion Sort)	105
		সিলেকশন সর্ট	106
		বাবল সর্ট (Bubble Sort)	107
		কুইক সর্ট (Quick Sort)	111
		মার্জ সর্ট (Merge Sort)	112
		র‍্যাডিক্স সর্ট	113
		Tower of Hanoi	114
		Abstract Data Type, Linked List	117
		এক্সপ্রেশনের ধরন	119
		Hashing, Hash Indices	120
		Trees	125
		Huffman Encoding Technique	131
		বি-ট্রি (B-Tree)	132
		বি+ ট্রি	133
		Graphs	135
		Shortest Path Problems	136
		Divide and Conquer	140
		Greedy Choice Property	141
		Fractional Knapsack Problem	142
		Activity Selection Problem	144

		Dynamic Programming (DP)	147
		Principle of Optimality, LCS	148
		Viterbi Algorithm	149
		SELF TEST	153
4	Chapter-4: Web Technology	Webpage, Website, Server, Hosting	155
		ব্রাউজার, সার্চ ইঞ্জিন, URL, আই পি অ্যাড্রেস	156
		ওয়েব সাইটের প্রকার ভেদ(স্ট্যাটিক ও ডায়নামিক ওয়েব সাইট	162
		HTML & Tag	167
		Making Table	174
		Making List	175
		হাইপারলিংক	177
		Colour Code	182
		ফর্ম তৈরি	182
		অডিও ও ভিডিও সংযোজন, MVC ফ্রেমওয়ার্ক	183
		Web Security, Security Threat	185
		ওয়েব সার্ভিস	186
		DoS, Cross-Site Scripting	190
		JavaScript	192
		CSS	194
		Secure Sockets Layer	195
		Common Gateway Interface, W3C	196
		গুরুত্বপূর্ণ বহুনির্বাচনি	198
		SELF TEST	203
5	Chapter-5: Operating System	অপারেটিং সিস্টেম	205
		Hardware concepts related to OS	209
		অপারেটিং সিস্টেমে প্রসেস	212
		Distributed Processing	213
		UNIX process control	215
		সিগন্যাল (Signals), পাইপস (Pipes)	216
		Job and processor scheduling, scheduling algorithms, process hierarchies.	220
		Semaphores (সেমাফোর)	223
		Critical regions, Conditional Critical Regions, Monitors, Ada Tasks	227
		Memory organization and management, storage allocation.	230
		পেজ রিপ্লেসমেন্ট স্ট্র্যাটেজি	232
		অপারেটিং সিস্টেমে Swapping (সোয়াপিং)	232
		ফাইল অর্গানাইজেশন ও ফাইল ডেসক্রিপ্টর	235
		SELF TEST	239
6		ডেটাবেজ ম্যানেজমেন্ট সিস্টেম	241

	Chapter-6: Database Management System	E-R Diagram	242
		ডেটা হায়ারার্কি	243
		কী (Key)	244
		রিলেশনাল ডেটাবেজ ম্যানেজমেন্ট সিস্টেম	244
		ফেইলিওর	248
		ডেটাবেজ রিলেশন	250
		ডেটাবেজের ফিল্ডের ডেটা টাইপ	256
		সটিং	257
		ইন্ডেক্সিং, হ্যাশিং	258
		রিলেশনশীপ ডিগ্রি, কুয়েরি	263
		এসকিউএল (SQL)	264
		এগ্রিগেট ফাংশন, ডেটাবেজ মডিফিকেশন	270
		অথরাইজেশন, নরমালাইজেশন	271
		ডেটা সিকিউরিটি	272
		ডেটা এনক্রিপশন	272
		পাবলিক কী (Public Key) ও প্রাইভেট কী (Private Key)	273
		প্যারালাল প্রসেসিং, ইন্টারফেসিং	278
		ডিস্ট্রিবিউটেড অপারেটিং সিস্টেম	279
		ডেটা মডেল	280
		ডেটা ডিকশনারি	280
		ডেটাবেজ স্কিমা	281
		B-Tree (বি-ট্রি), B+ ট্রি	285
		ডেটা ওয়ারহাউসিং, ডেটা মাইনিং, স্পিডআপ, স্কেলআপ	286
		ডেটা ফ্লাগমেন্টেশন	287
		গুরুত্বপূর্ণ MCQ	289
		SELF TEST	294
7	Chapter-7: Data Communication System and Networking	উপাত্ত বা ডেটা, ইনফরমেশন	297
		ডেটা ট্রান্সমিশন ব্যান্ডউইথ	298
		ডেটা ট্রান্সমিশন মেথড	298
		ডেটা ট্রান্সমিশন মোড	304
		তার মাধ্যম (Wirebase)	305
		তারবিহীন মাধ্যম	309
		ওয়্যারলেস কমিউনিকেশন সিস্টেম	311
		সেলুলার মোবাইল সিস্টেম	315
		মোবাইল যোগাযোগ	315
		কম্পিউটার নেটওয়ার্ক	319
		নেটওয়ার্ক ডিভাইস	321
		নেটওয়ার্ক টপোলজি	327
		ক্লাউড কম্পিউটিং	329
		OSI মডেল, TCP/IP	333

		ইন্টারনেট	334
		ইমেইল (Email)	335
		ই-কমার্স	336
		রাউটিং অ্যালগরিদম	337
		মাল্টিপ্লেস্কার	338
		গুরুত্বপূর্ণ MCQ	343
		SELF TEST	351
8	Chapter-8: Question Bank	High Voltage MCQ	353
		BCS Question Bank(10 <sup>th</sup> to 49 <sup>th</sup> )	369
		Bank Question Bank	390
		Model Test	408

কোনো টপিকস সহজেই বুঝতে চাইলে সাবস্কাইব করুন আমাদের ইউটিউব চ্যানেল।



Follow us to updates and any query  
<https://www.synchronousboi.com>

লেখকের লিখিত অনুমতি ছাড়া এই বই বা এর যেকোনো অংশ হুবহু, আংশিক, অনুবাদ, পুনর্মুদ্রণ, ফটোকপি, স্ক্যান, ডিজিটাল কপি, অডিও বা ভিডিও আকারে—কোনোভাবেই এবং কোনো মাধ্যমে ব্যবসায়িক উদ্দেশ্যে ব্যবহার, প্রচার বা বিতরণ করা সম্পূর্ণ নিষিদ্ধ।

## Chapter-1

### Structured and Object-Oriented Programming (OOP) Concept

#### প্রোগ্রাম ডিজাইন মডেলঃ

সহজ উপায়ে কার্যকরী প্রোগ্রাম তৈরির জন্য যে বিশেষ নীতিমালা বা পদ্ধতি অনুসরণ করা হয় তাকে প্রোগ্রাম ডিজাইন মডেল বলে। কয়েকটি জনপ্রিয় প্রোগ্রাম ডিজাইন মডেল-

- ১। স্ট্রাকচার্ড বা প্রসিডিউর প্রোগ্রামিং মডেল।
- ২। অবজেক্ট ওরিয়েন্টেড প্রোগ্রামিং(OOP) মডেল।
- ৩। ভিজুয়াল প্রোগ্রামিং মডেল।
- ৪। ইভেন্ট ড্রাইভেন প্রোগ্রামিং মডেল।

#### স্ট্রাকচার্ড প্রোগ্রামিং:

১৯৬৬ সালে স্ট্রাকচার্ড প্রোগ্রামিং এর প্রথম ধারণা দেন Corrado Bohm এবং Guiseppe Jacopini। এই দুই গণিতবিদ ব্যাখ্যা করেন যে, যেকোন প্রোগ্রাম শুধুমাত্র তিনটি স্ট্রাকচার যেমন- decisions, sequences, এবং loops এর সাহায্যে লেখা যায়। পরবর্তিতে ১৯৭০ সালে Edsger W.Dijkstra ব্যপকভাবে ব্যবহৃত স্ট্রাকচার্ড প্রোগ্রামিং পদ্ধতি উন্নয়ন করেন, যেখানে একটি সমস্যাকে বিভিন্ন ছোট ছোট মডিউল বা অংশে ভাগ করে একটি বড় সমস্যার সমাধান করা হয়। প্রতিটি মডিউলকে ফাংশন বলা হয়।

উদাহরণঃ সি, কোবল, প্যাসকেল, ফরট্রান, কিউবেসিক ইত্যাদি প্রোগ্রামিং ভাষায় স্ট্রাকচার্ড প্রোগ্রামিং ডিজাইন অনুসরণ করে প্রোগ্রাম লেখা যায় এজন্য এই প্রোগ্রামিং ভাষা গুলোকে স্ট্রাকচার্ড প্রোগ্রামিং ভাষা বলা হয়।

#### অবজেক্ট ওরিয়েন্টেড প্রোগ্রামিংঃ

অবজেক্ট ওরিয়েন্টেড প্রোগ্রামিং হল এমন একটি প্রোগ্রামিং পদ্ধতি যা স্ট্রাকচার্ড প্রোগ্রামিং এর সুবিধার পাশাপাশি অতিরিক্ত বিশেষ কিছু সুবিধা যেমন- এনক্যাপ্সুলেশন, পলিমরফিজম ও ইনহেরিটেন্স প্রভৃতি ফিচার ব্যবহার করে প্রোগ্রাম লেখার সুবিধা প্রদান করে।

উদাহরণঃ জাভা, পাইথন, সি++ সি# ইত্যাদি হল অবজেক্ট ওরিয়েন্টেড প্রোগ্রামিং ভাষা। অবজেক্ট ওরিয়েন্টেড প্রোগ্রামিং ভাষার বৈশিষ্ট্যসমূহঃ

**অবজেক্টঃ** অবজেক্ট ওরিয়েন্টেড প্রোগ্রামিং ভাষায় যেকোন ব্যক্তি বা বস্তুকে অবজেক্ট বলা হয়। যেমন – একটি গাড়ি কে অবজেক্ট বলা যায়। প্রতিটি অবজেক্ট এর কিছু বৈশিষ্ট্য(attribute) ও আচরণ(behavior) থাকে। যেমন একটি গাড়ির কালার, মডেল ইত্যাদি হল বৈশিষ্ট্য, আবার গাড়িটি সামনে চলতে পারে এবং পিছনে চলতে পারে এগুলো হল আচরণ।

**ক্লাসঃ** অবজেক্ট ওরিয়েন্টেড প্রোগ্রামিং ভাষায় ক্লাস হল ভেরিয়েবল ও মেথডের সমন্বয়ে একটি টেমপ্লেট বা ব্ল-প্রিন্ট যা কোন অবজেক্ট এর বৈশিষ্ট্য(attribute) ও আচরণ(behavior) উপস্থাপনের জন্য তৈরি করা হয়।

**এনক্যাপ্সুলেশনঃ** অবজেক্ট এর বৈশিষ্ট্য(attribute) ও আচরণ(behavior) কে একত্র করে ক্লাস তৈরি করাকে বলা হয় এনক্যাপ্সুলেশন।

**পলিমরফিজমঃ** পলিমরফিজম মানে হল বহুরূপ। একাদিক কোড মডিউলের নাম এক হলেও ভিন্ন ভিন্ন রূপ থাকতে পারে, এক্ষেত্রে কোন মডিউলটি কাজ করবে তা নির্ভর করে ডেটা পাঠানোর উপর।

**ইনহেরিটেন্সঃ** অবজেক্ট ওরিয়েন্টেড প্রোগ্রামিং ভাষায় ইনহেরিটেন্স এমন একটি ফিচার, যার কারণে একটি ক্লাসের বৈশিষ্ট্য অপর একটি ক্লাস ব্যবহার করতে পারে, একে বলা হয় ইনহেরিট করা। যে ক্লাস কে ইনহেরিট করা হয় তাকে বলে বেজ ক্লাস এবং যে ক্লাস অন্য ক্লাসকে ইনহেরিট করে তাকে বলে ডিরাইভড ক্লাস।

#### ভিজুয়াল প্রোগ্রামিংঃ

ভিজুয়াল প্রোগ্রামিং হল এমন একটি প্রোগ্রামিং পদ্ধতি যেখানে বিভিন্ন চিত্রভিত্তিক নির্দেশ বা কমান্ড ব্যবহার করে প্রোগ্রাম রচনা করা হয়। স্ট্রাকচার্ড বা অবজেক্ট ওরিয়েন্টেড প্রোগ্রামিং এর উপর ভিত্তি করেই ভিজুয়াল প্রোগ্রামিং মডেল তৈরি। মাইক্রোসফট কোম্পানির তৈরিকৃত ভিজুয়াল বেসিক হল প্রথম ভিজুয়াল প্রোগ্রামিং মডেল।

#### ইভেন্ট ড্রাইভেন প্রোগ্রামিংঃ

সকল ভিজুয়াল প্রোগ্রাম হচ্ছে ইভেন্ট ড্রাইভেন প্রোগ্রাম। এই মডেলে কী-বোর্ডের কোন কী চাপ দেওয়া, কোন বিশেষ কন্ট্রোলার উপর মাউস ক্লিক করা ইত্যাদি কাজ গুলো হচ্ছে এক একটি ইভেন্ট। প্রতিটি ইভেন্টের জন্য পৃথক পৃথক কোড মডিউল থাকে। ব্যবহারকারী যখন কোন ইভেন্ট একটিভ করেন তখন ঐ ইভেন্টের জন্য নির্ধারিত কোড মডিউলটি নির্বাহ হয়।

**প্রোগ্রাম ডিজাইন মডেল থেকে গুরুত্বপূর্ণ টিপস**

১. প্রোগ্রামের গঠন রীতিনীতিকে বলা হয়- প্রোগ্রাম মডেল।
২. প্রথম চিত্রভিত্তিক প্রোগ্রামিং মডেল হলো- ভিজুয়্যাল বেসিক।
৩. OOP বা অবজেক্ট ওরিয়েন্টেড প্রোগ্রামিং এর বৈশিষ্ট্য- ইনহেরিটেন্স, এনক্যাপসুলেশন, পলিমরফিজম।
৪. পলিমরফিজম অর্থ হচ্ছে- বহুরূপ।
৫. OOP প্রোগ্রামিং ভাষার উদাহরণ হলো- C++, Java, C# ইত্যাদি।
৬. জলপ্রপাত মডেলে প্রোগ্রাম ডিজাইন ও কোডিং এ সময় ব্যয় হয়- 30 থেকে 40% সময়।

**প্রোগ্রামের ধারণাঃ**

কত গুলো নির্দেশনার সমষ্টিকে প্রোগ্রাম বলা হয়। কম্পিউটার প্রোগ্রামও কিছু নির্দেশনার সমষ্টি।

**প্রোগ্রামিং ভাষাঃ**

কম্পিউটার প্রোগ্রাম তৈরির জন্য কিছু বর্ণ, চিহ্ন, অঙ্ক, বিশেষ চিহ্ন দিয়ে প্রোগ্রাম তৈরির কৌশলকেই প্রোগ্রামিং ভাষা বলে।

**প্রোগ্রামিং ভাষার স্তরঃ** কম্পিউটার প্রোগ্রামিং ভাষাকে ৫ টি স্তরে ভাগ করা হয়েছে-

- ১। প্রথম প্রজন্মঃ মেশিন/ যান্ত্রিক ভাষা (১৯৪৫)
- ২। দ্বিতীয় প্রজন্মঃ অ্যাসেম্বলি ভাষা (১৯৫০)
- ৩। তৃতীয় প্রজন্মঃ উচ্চস্তরের ভাষা (১৯৬০)
- ৪। চতুর্থ প্রজন্মঃ অতি উচ্চস্তরের ভাষা (১৯৭০)
- ৫। পঞ্চম প্রজন্মঃ স্বাভাবিক বা ন্যাচারাল ভাষা (১৯৮০)

**প্রোগ্রাম রচনার ভিত্তিতে ভাষাকে ২ ভাগে ভাগ করাঃ**

- ১। নিম্ন স্তরের ভাষাঃ যান্ত্রিক ও অ্যাসেম্বলি ভাষা।
- ২। উচ্চস্তরের ভাষাঃ উচ্চস্তরের, অতি উচ্চস্তরের ও ন্যাচারাল ভাষা।

**মেশিন বা যান্ত্রিক ভাষাঃ**

যে ভাষায় 0 এবং 1 ব্যবহার করে প্রোগ্রাম বা কোড লেখা হয় তাকে মেশিন বা যান্ত্রিক ভাষা বলে। কম্পিউটারের নিজস্ব ভাষা হচ্ছে মেশিন ভাষা। মেশিন ভাষায় লেখা কোনো প্রোগ্রামকে অবজেক্ট বা বস্তু প্রোগ্রাম বলা হয়।

**মেশিন ভাষার সুবিধাঃ**

- ১। মেশিন ভাষার সবচেয়ে বড় সুবিধা হচ্ছে এই ভাষায় লেখা প্রোগ্রাম কম্পিউটার সরাসরি বুঝতে পারে।
- ২। কোনো প্রকার অনুবাদক প্রোগ্রামের প্রয়োজন হয় না।
- ৩। দ্রুত কাজ করে।
- ৪। মেশিন ভাষায় লেখা প্রোগ্রামে অতি অল্প মেমোরি প্রয়োজন হয়।

**মেশিন ভাষার অসুবিধাঃ**

- ১। শুধু ০ ও ১ ব্যবহার করা হয় বলে মেশিন ভাষা শেখা কষ্টকর ও প্রোগ্রাম লেখাও কষ্টসাধ্য।
- ২। প্রোগ্রাম লিখতে প্রচুর সময় লাগে।
- ৩। এই ভাষায় লেখা কোনো প্রোগ্রাম সহজে বোঝা যায় না।
- ৪। এই ভাষায় প্রোগ্রাম লিখলে ভুল হওয়ার সম্ভাবনা খুব বেশি থাকে।
- ৫। ভুল হলে তা বের করা এবং ভুল-ত্রুটি দূর করা অনেক কঠিন।
- ৬। এই ভাষা হলো যন্ত্র নির্ভর ভাষা যা এক ধরনের যন্ত্রে জন্য লেখা প্রোগ্রাম অন্য ধরনের যন্ত্রে ব্যবহার করা যায় না।
- ৭। এই ভাষায় প্রোগ্রাম রচনার ক্ষেত্রে কম্পিউটারের অভ্যন্তরীণ সংগঠন ভালোভাবে জানতে হয়।

**অ্যাসেম্বলি ভাষাঃ**

মেশিন ভাষায় নির্দেশনা দিতে বা প্রোগ্রাম লিখতে অনেক পরিশ্রম হত। ভুল সংশোধন করা কষ্টকর ছিল। এই সমস্যা সমাধানের জন্য সাংকেতিক চিহ্ন(ADD, SUM, DIV, MUL, INP, OUT, CLR ইত্যাদি) দিয়ে নতুন ভাষা তৈরি করা হয়। যে ভাষায় বিভিন্ন সংকেত বা নেমোনিক ব্যবহার করে প্রোগ্রাম লেখা হয় তাকে অ্যাসেম্বলি ভাষা বলে। অ্যাসেম্বলি ভাষায় প্রোগ্রাম লেখার জন্য ০ ও ১ ব্যবহার না করে বিভিন্ন সংকেত ব্যবহার করা হয়। এই সংকেতকে বলে সাংকেতিক কোড (Symbolic Code) বা নেমোনিক (Nemonic) কোড এবং এটি সর্বোচ্চ পাঁচটি লেটারের সমন্বয়ে হয়।



অ্যাসেম্বলি ভাষায় প্রতিটি নির্দেশের চারটি অংশ থাকে। যথা-

১. লেবেল (সাংকেতিক ঠিকানা)
২. অপ-কোড (নেমোনিক কোড থাকে)
৩. অপারেন্ড (অপকোড বা নেমোনিক কোড যাদের উপর কাজ করে- A, B, X, Y ইত্যাদি বর্ণ হয়ে থাকে)
৪. কমেন্ট (প্রোগ্রামের অংশ নয়)

অ্যাসেম্বলি ভাষার সুবিধা:

- ১। ভাষা সহজে বুঝা যায় এবং এই ভাষায় প্রোগ্রাম রচনা করা যান্ত্রিক ভাষার তুলনায় অনেক সহজ।
- ২। প্রোগ্রাম রচনা করতে সময় এবং শ্রম কম লাগে।
- ৩। প্রোগ্রামের ত্রুটি বের করে তা সমাধান করা এবং প্রোগ্রাম পরিবর্তন করা সহজ।

অ্যাসেম্বলি ভাষার অসুবিধা:

- ১। এক ধরনের কম্পিউটারের জন্য লেখা প্রোগ্রাম অন্য কম্পিউটারে ব্যবহার করা যায় না। অর্থাৎ যন্ত্র নির্ভর ভাষা।
- ২। প্রোগ্রাম রচনার সময় প্রোগ্রামারকে মেশিন সম্পর্কে ধারণা থাকতে হয়।
- ৩। অনুবাদক প্রোগ্রামের প্রয়োজন হয়।

### উচ্চস্তরের ভাষাঃ

উচ্চস্তরের ভাষা হলো সেই সকল ভাষা যা মানুষের বোধগম্য এবং মানুষের ভাষার কাছাকাছি। যেমন- উচ্চস্তরের ভাষা ইংরেজি ভাষার সাথে মিল আছে এবং এই প্রোগ্রামিং ভাষা যন্ত্র নির্ভর নয়, এই জন্য এসব ভাষাকে উচ্চস্তরের ভাষা বলা হয়। এটি মানুষের জন্য বুঝা খুব সহজ কিন্তু কম্পিউটার সরাসরি বুঝতে পারে না বলে অনুবাদক প্রোগ্রামের সাহায্যে একে মেশিন ভাষায় রূপান্তর করে নিতে হয়। এটি তৃতীয় প্রজন্মের ভাষা।

উচ্চস্তরের ভাষার সুবিধা

- ১। উচ্চস্তরের ভাষা মানুষের ভাষা যেমন- ইংরেজি ভাষার কাছাকাছি। তাই শেখা সহজ ফলে এই ভাষা ব্যবহার করে প্রোগ্রাম লেখা সহজ ও লিখতে সময় কম লাগে।
- ২। এতে ভুল হওয়ার সম্ভাবনা কম থাকে এবং ডিবাগিং সহজ।
- ৩। প্রোগ্রাম লেখার জন্য কম্পিউটারের ভেতরের সংগঠন সম্পর্কে ধারণা থাকার প্রয়োজন নেই।
- ৪। এক কম্পিউটারের জন্য লেখা প্রোগ্রাম অন্য মডেলের কম্পিউটারে চলে। অর্থাৎ যন্ত্র নির্ভর নয়।

উচ্চস্তরের ভাষার অসুবিধা

- ১। এই ভাষার সাহায্যে কম্পিউটারের সাথে সরাসরি যোগাযোগ করা যায় না।
- ২। এই ভাষায় লেখা প্রোগ্রামকে অনুবাদ করে কম্পিউটারকে বুঝিয়ে দিতে হয়। অনুবাদক প্রোগ্রামের প্রয়োজন হয়
- ৩। বেশি মেমোরি প্রয়োজন হয়।

উচ্চস্তরের ভাষার ব্যবহার

- ১। বড় অ্যাপ্লিকেশন তৈরির কাজে।
- ২। জটিল গাণিতিক হিসাব-নিকাশে ব্যবহৃত সফটওয়্যার তৈরি করতে।
- ৩। অ্যাপ্লিকেশন প্যাকেজ সফটওয়্যার তৈরিতে
- ৪। বিভিন্ন ধরনের অটোমেটিক প্রসেস কন্ট্রলের কাজে।

### জনপ্রিয় কিছু উচ্চস্তরের প্রোগ্রামিং ভাষার পরিচিতিঃ

**সি (C):** 'সি' প্রোগ্রামিং ভাষা একটি স্ট্রাকচার্ড বা প্রোসিডিউর প্রোগ্রামিং ভাষা। মিদ লেভেল ল্যাঙ্গুয়েজ হিসেবে 'সি' অত্যন্ত জনপ্রিয়। ১৯৭২ সালে আমেরিকার বেল ল্যাবরেটরিতে ডেনিস রিচি 'সি' ভাষা উদ্ভাবন করেন। ডেনিস রিচিকে 'সি' প্রোগ্রামিং ভাষার জনক বলা হয়। 'সি' নামটা এসেছে মার্টিন রিচার্ডস (Martins Richards) এর উদ্ভাবিত বিসিপিএল (BCPL-Basic Combined Programming Language) ভাষা থেকে। BCPL সংক্ষেপে B নামে পরিচিত ছিল। পরে B এর উন্নয়নের ফলে C এর বিকাশ ঘটে।

**সি++(C++):** ১৯৮৩ সালে Bjarne Stroustrup বেল ল্যাবরেটরিতে C ভাষার বৈশিষ্ট্যের সাথে অবজেক্ট ওরিয়েন্টেড প্রোগ্রামিং এর বৈশিষ্ট্য যুক্ত করে নতুন এক প্রোগ্রামিং ভাষা তৈরি করেন যা C++ নামে পরিজায়া। এই ভাষাকে অবজেক্ট ওরিয়েন্টেড প্রোগ্রামিং ভাষা বলা হয়। এই ভাষার সাহায্যে বিভিন্ন সিস্টেম সফটওয়্যার, অ্যাপ্লিকেশন সফটওয়্যার, ডিভাইস ড্রাইভার ইত্যাদি তৈরি করা যায়।

**জাভাঃ** সান মাইক্রোসিস্টেম কোম্পানি জাভা প্রোগ্রামিং ভাষাটি তৈরি করেন। James Gosling কে জাভা প্রোগ্রামিং ভাষার জনক বলা হয়। জাভা প্রোগ্রামিং ভাষা আবিষ্কার হয় ১৯৯৫ সালে। এই প্রোগ্রামিং ভাষা ব্যবহার করে ওয়েব অ্যাপ্লিকেশন, ডেস্কটপ অ্যাপ্লিকেশন, মোবাইল অ্যাপ্লিকেশন ইত্যাদি তৈরি করা যায়।

**পাইথনঃ** ১৯৯১ সালে Guido Van Rossum পাইথন প্রোগ্রামিং ভাষা তৈরি করেন। এই ভাষা একই সাথে অবজেক্ট ওরিয়েন্টেড ও স্ট্রাকচার্ড প্রোগ্রামিং ভাষার বৈশিষ্ট্য সাপোর্ট করে। ক্লাউড ভিত্তিক ওয়েব অ্যাপ্লিকেশন, ডেটা এনালাইসিস ও মেশিন লার্নিং অ্যাপ্লিকেশন তৈরিতে এই ভাষা ব্যবহার অত্যন্ত জনপ্রিয়।

**ভিজুয়াল বেসিকঃ** ভিজুয়াল বেসিক একটি ইভেন্ট ড্রাইভেন প্রোগ্রামিং ডিজাইন ভিত্তিক ভাষা যা মাইক্রোসফট তৈরি করে।

**অ্যালগলঃ** এর পূর্ণনাম Algorithmic Language। এটি ব্যবহৃত হতো মূলত বৈজ্ঞানিক গবেষণায় বিভিন্ন সমস্যার সমাধানে।

**ফোরট্রানঃ** Fortran এর অর্থ Formula Translator যা উচ্চস্তরের প্রোগ্রামিং ভাষাগুলোর মধ্যে সবচেয়ে আদিমতম ভাষা। IBM এর গবেষক জন বাকাস IBM মেইনফ্রেম কম্পিউটার এর জন্য এটি তৈরি করেন।

**ওরাকলঃ** ওরাকল একটি RDBMS(Relational Database Management System) সফটওয়্যার যা ওরাকল কর্পোরেশন তৈরি করে। এটি ডেটাবেজ সফটওয়্যারগুলোর মধ্যে সবচেয়ে জনপ্রিয়।

#### **মধ্যমস্তরের ভাষাঃ**

যে প্রোগ্রামিং ভাষায় নিম্নস্তরের ভাষার সুবিধা যেমন- বিট পর্যায়ে প্রোগ্রামিং বা সিস্টেম সফটওয়্যার এর মাধ্যমে হার্ডওয়্যার নিয়ন্ত্রণ এবং উচ্চস্তরের ভাষার সুবিধা যেমন- অ্যাপ্লিকেশন সফটওয়্যার তৈরি করা যায় তাকে মধ্যম স্তরের ভাষা বলা হয়। মধ্যম স্তরের ভাষার উদাহরণ হল – C, Forth, Dbase, WordStar ইত্যাদি।

#### **চতুর্থ প্রজন্মের ভাষা (4<sup>th</sup> Generation Language-4GL):**

4GL এর পূর্ণরূপ Fourth Generation Language। চতুর্থ প্রজন্মের ভাষাকে অতি উচ্চ স্তরের ভাষা বলা হয়। চতুর্থ প্রজন্মের ভাষা হলো ডেটাবেজ সংক্রান্ত ভাষা। অর্থাৎ এই প্রজন্মের ভাষার সাহায্যে ডেটাবেজ তৈরি, আপডেট, ডিলেট সহ ডেটাবেজ সম্পর্কিত সকল কাজ সম্পাদন করা যায়। এই প্রজন্মের ভাষার উদাহরণ হল SQL, Oracle, FOCUS, INTELLECT, Visual BASIC ইত্যাদি।

#### **পঞ্চম প্রজন্মের ভাষা (5<sup>th</sup> Generation Language-5GL):**

5GL এর পূর্ণরূপ Fifth Generation Language। পঞ্চম প্রজন্মের ভাষাকে স্বাভাবিক ভাষা (Natural Language) ও বলা হয়। Artificial Intelligence বা কৃত্রিম বুদ্ধিমত্তা নির্ভর যন্ত্র তৈরিতে এই প্রজন্মের ভাষা ব্যবহৃত হয়। পঞ্চম প্রজন্মের ভাষায় লেখা প্রোগ্রামকে মেশিন ভাষায় রূপান্তরের জন্য ইন্টেলিজেন্ট কম্পাইলার ব্যবহৃত হয়। এই প্রজন্মের ভাষা ব্যবহার করে মানুষ যন্ত্রকে মৌখিক নির্দেশ দিতে পারে। পঞ্চম প্রজন্মের ভাষার উদাহরণ হচ্ছে PROLOG (PROgramming LOGic), LISP, Mercury ইত্যাদি।

### **প্রোগ্রামিং ভাষা থেকে গুরুত্বপূর্ণ টিপস**

১. সফটওয়্যার তৈরির জন্য প্রয়োজন হলো প্রোগ্রামিং ভাষা।
২. কম্পিউটার প্রসেসর এক্সিকিউট করতে পারে ইনস্ট্রাকশন সেট।
৩. প্রোগ্রাম লেখা কঠিন ও সময় সাপেক্ষ মেশিন ও অ্যাসেম্বলি ভাষা।
৪. প্রোগ্রাম লেখা ও পড়া সহজ উচ্চস্তরের ভাষায়।
৫. লাইব্রেরি লিংক থাকে ভিজুয়াল বেসিক ভাষায়।

## MCQ

১. সহজ উপায়ে কার্যকরী প্রোগ্রাম তৈরির জন্য যে নীতিমালা অনুসরণ করা হয় তাকে কী বলে?

- A. প্রোগ্রাম কোড B. প্রোগ্রাম ডিজাইন মডেল  
C. অ্যালগরিদম D. ফ্লোচার্ট

Ans: B

২. নিচের কোনটি প্রোগ্রাম ডিজাইন মডেল নয়?

- A. স্ট্রাকচার্ড B. অবজেক্ট ওরিয়েন্টেড  
C. ভিজুয়াল D. মেশিন ভাষা

Ans: D

৩. স্ট্রাকচার্ড প্রোগ্রামিং এর প্রথম ধারণা দেন কারা?

- A. Dennis Ritchie ও Bjarne Stroustrup  
B. Corrado Bohm ও Guiseppe Jacopini  
C. James Gosling ও Guido van Rossum  
D. Edsger W. Dijkstra ও Dennis Ritchie

Ans: B

৪. স্ট্রাকচার্ড প্রোগ্রামিং এ কয়টি মৌলিক স্ট্রাকচার রয়েছে?

- A. 2 B. 3 C. 4 D. 5

Ans: B

৫. স্ট্রাকচার্ড প্রোগ্রামিং এর তিনটি স্ট্রাকচার হলো—

- A. Class, Object, Method  
B. Input, Output, Process  
C. Sequence, Decision, Loop  
D. Compile, Run, Debug

Ans: C

৬. স্ট্রাকচার্ড প্রোগ্রামিং পদ্ধতি জনপ্রিয় করেন—

- A. Corrado Bohm B. James Gosling  
C. Edsger W. Dijkstra D. Guido van Rossum

Ans: C

৭. স্ট্রাকচার্ড প্রোগ্রামিং এ সমস্যাকে ছোট অংশে ভাগ করে সমাধান করা হয়, একে বলে—

- A. Encapsulation B. Modularization  
C. Inheritance D. Polymorphism

Ans: B

৮. নিচের কোনটি স্ট্রাকচার্ড প্রোগ্রামিং ভাষা?

- A. Java B. Python C. C D. C#

Ans: C

৯. OOP এর পূর্ণরূপ কী?

- A. Object Oriented Process  
B. Open Oriented Program  
C. Object Oriented Programming  
D. Open Object Programming

Ans: C

১০. নিচের কোনটি OOP এর বৈশিষ্ট্য নয়?

- A. Encapsulation B. Polymorphism  
C. Inheritance D. Compilation

Ans: D

১১. পলিমরফিজম শব্দের অর্থ কী?

- A. একরূপ B. বহুরূপ C. দূত কাজ D. পুনরাবৃত্তি

Ans: B

১২. অবজেক্ট বলতে কী বোঝায়?

- A. শুধু ডেটা  
B. শুধু ফাংশন  
C. বাস্তব জগতের কোনো ব্যক্তি বা বস্তু  
D. কোডের অংশ

Ans: C

১৩. ক্লাস হলো—

- A. অবজেক্টের বাস্তব রূপ  
B. ভেরিয়েবল ও মেথডের ব্লু-প্রিন্ট  
C. শুধু ডেটা সংরক্ষণ  
D. শুধু ফাংশনের সমষ্টি

Ans: B

১৪. এক ক্লাসের বৈশিষ্ট্য অন্য ক্লাস ব্যবহার করলে তাকে কী বলে?

- A. Encapsulation B. Polymorphism  
C. Inheritance D. Abstraction

Ans: C

১৫. OOP ভাষার উদাহরণ কোনটি?

- A. COBOL B. FORTRAN C. Java D. Assembly

Ans: C

১৬. চিত্রভিত্তিক নির্দেশ ব্যবহার করে প্রোগ্রাম লেখাকে কী বলে?

- A. Structured Programming  
B. Visual Programming  
C. Machine Programming  
D. Procedural Programming

Ans: B

১৭. প্রথম ভিজুয়াল প্রোগ্রামিং ভাষা কোনটি?

- A. Java B. Python C. Visual Basic D. C++

Ans: C

১৮. সব ভিজুয়াল প্রোগ্রাম মূলত কোন মডেলের উপর ভিত্তি করে?

- A. Compiler Driven B. Event Driven  
C. Procedure Driven D. Machine Driven

Ans: B

**অনুবাদক প্রোগ্রামঃ**

যে প্রোগ্রাম উৎস(Source) প্রোগ্রামকে বস্তু(Object) প্রোগ্রামে রূপান্তর করে তাকে অনুবাদক প্রোগ্রাম বলে। মেশিন ভাষায় লেখা প্রোগ্রামকে বলা হয় বস্তু প্রোগ্রাম (Object Program) এবং অন্য যেকোনো ভাষায় লেখা প্রোগ্রামকে বলা হয় উৎস প্রোগ্রাম (Source program)। অনুবাদক প্রোগ্রাম উৎস প্রোগ্রামকে ইনপুট হিসেবে নেয় এবং বস্তু প্রোগ্রামকে আউটপুট হিসেবে দেয়। প্রোগ্রাম অনুবাদের সময় উৎস প্রোগ্রামে যদি কোন ভুল থাকে, তবে তা সংশোধন করার জন্য ব্যবহারকারীকে Error Message দেয়।

**অনুবাদক প্রোগ্রামের প্রকারভেদ-**

- ১। অ্যাসেম্বলার(Assembler)
- ২। কম্পাইলার(Compiler)
- ৩। ইন্টারপ্রেটার(Interpreter)

**অ্যাসেম্বলারঃ** অ্যাসেম্বলার হলো এক ধরনের অনুবাদক প্রোগ্রাম যা অ্যাসেম্বলি ভাষায়

লেখা প্রোগ্রামকে মেশিন ভাষায় রূপান্তর করে।

**কম্পাইলারঃ** কম্পাইলার হলো এক ধরনের অনুবাদক প্রোগ্রাম যা উচ্চস্তরের ভাষায় লেখা প্রোগ্রামকে একসাথে মেশিন বা যান্ত্রিক ভাষায় রূপান্তর করে।

**ইন্টারপ্রেটারঃ** ইন্টারপ্রেটারও কম্পাইলারের মতো এক ধরনের অনুবাদক প্রোগ্রাম যা উচ্চস্তরের ভাষায় লেখা প্রোগ্রামকে লাইন বা লাইন মেশিন বা যান্ত্রিক ভাষায় রূপান্তর করে।

**কম্পাইলার এবং ইন্টারপ্রেটারের মধ্যে পার্থক্যঃ**

কম্পাইলার	ইন্টারপ্রেটার
উচ্চস্তরের প্রোগ্রামিং ভাষায় লেখা একটি সম্পূর্ণ প্রোগ্রামকে একসাথে অনুবাদ করে।	উচ্চস্তরের প্রোগ্রামিং ভাষায় লেখা একটি প্রোগ্রামকে লাইন বাই লাইন অনুবাদ করে।
সবগুলো ভুল একসাথে প্রদর্শন করে।	ভুল গুলো লাইন বাই লাইন প্রদর্শন করে।
ডিবাগিং ও টেস্টিং ধীর গতি সম্পন্ন।	ডিবাগিং ও টেস্টিং দ্রুত গতি সম্পন্ন।
একবার অনুবাদ করলে পরে আর অনুবাদ করার প্রয়োজন হয় না।	প্রতিবার কাজের পূর্বে অনুবাদ করার প্রয়োজন হয়ে থাকে।
কম্পাইলার দ্বারা একটি প্রোগ্রাম অনুবাদ করলে পূর্ণাঙ্গ যান্ত্রিক প্রোগ্রামে রূপান্তরিত হয়।	ইন্টারপ্রেটার দ্বারা একটি প্রোগ্রাম অনুবাদ করলে পূর্ণাঙ্গ যান্ত্রিক প্রোগ্রামে রূপান্তরিত হয় না।
কম্পাইলারে কাজের প্রক্রিয়া জটিল।	ইন্টারপ্রেটার কাজের প্রক্রিয়া কিছুটা সহজ।
প্রধান মেমোরিতে বেশি জায়গা প্রয়োজন।	কম জায়গা প্রয়োজন।

**প্রোগ্রাম সংগঠনঃ**

প্রতিটি পূর্ণাঙ্গ প্রোগ্রামের তিনটি অপরিহার্য অংশ থাকে, যা পারস্পরিক সম্পর্কের মাধ্যমে একটি পূর্ণাঙ্গ প্রোগ্রাম গঠিত হয়। যেমন-

- ১। **ইনপুট:** প্রতিটি প্রোগ্রামে প্রসেস বা প্রক্রিয়া করার জন্য ইনপুট নেওয়ার ব্যবস্থা থাকতে হবে।
- ২। **প্রসেস বা প্রক্রিয়া:** ব্যবহারকারীর কাছ থেকে ইনপুট নিয়ে প্রসেস বা প্রক্রিয়া করার ব্যবস্থা থাকতে হবে।
- ৩। **আউটপুট:** ইনপুট নিয়ে প্রসেস করে আউটপুট দেখানোর ব্যবস্থাও থাকতে হবে।

**একটি আদর্শ প্রোগ্রামের নিম্নোক্ত বৈশিষ্ট্য সমূহ থাকতে হয়ঃ**

- ১। প্রোগ্রাম অবশ্যই সহজ ও বোধগম্য হতে হবে, যাতে অন্যকোনো প্রোগ্রামার পরবর্তীতে আপডেট করতে পারে।
- ২। প্রোগ্রামটি রান করার জন্য সময় ও মেমোরি ন্যূনতম হতে হবে।
- ৩। প্রোগ্রাম সহজে সম্প্রসারণ যোগ্য হতে হবে।
- ৪। ডিবাগিং এবং টেস্টিং করা সহজতর হতে হবে।
- ৫। প্রোগ্রাম সহজে রক্ষণাবেক্ষণ যোগ্য হতে হবে।

একটি প্রোগ্রাম তৈরির মাধ্যমে সাধারণত একটি নির্দিষ্ট সমস্যার সমাধান করা হয়ে থাকে। তাই একটি প্রোগ্রাম তৈরি করার জন্য কতগুলো ধাপ অনুসরণ করলে সমস্যাটি সহজে সমাধান করা যায়।

**ধাপগুলো নিম্নোক্ত আলোচনা করা হল-**

- ১। সমস্যা নির্দিষ্টকরণ
- ২। সমস্যা বিশ্লেষণ

- ৩। প্রোগ্রাম ডিজাইন(অ্যালগরিদম ও ফ্লোচার্ট)
- ৪। প্রোগ্রাম উন্নয়ন বা কোডিং
- ৫। প্রোগ্রাম বাস্তবায়ন
- ৬। ডকুমেন্টেশন
- ৭। প্রোগ্রাম রক্ষণাবেক্ষণ

**ডিবাগিং:** প্রোগ্রামে যেকোনো ভুল চিহ্নিত করতে পারলে সেই ভুলকে বলা হয় বাগ (Bug)। উক্ত ভুল বা Bug কে সমাধান করাকে বলা হয় ডিবাগ (Debug)। অর্থাৎ প্রোগ্রামের ভুল খুঁজে বের করে তা সমাধান করার পদ্ধতিকে বলা হয় ডিবাগিং। একটি প্রোগ্রামে ৩ ধরনের ভুল থাকতে পারে।

**সিনটাক্স ভুল/ব্যাকরণগত ভুলঃ** প্রোগ্রামের মধ্যে প্রোগ্রামিং ভাষার ব্যাকরণগত যেসব ভুল থাকে তাকে বলা হয় সিনটাক্স ভুল। যেমন- বানান ভুল,কমা, ব্র্যাকেট ঠিকমতো না দেয়া, কোনো চলকের মান না জানানো প্রভৃতি। এসব ভুল সংশোধন করা খুবই সহজ, কারণ সিনটাক্স ভুল হলে অনুবাদক প্রোগ্রাম ভুলের বার্তা ছাপায়।

যেমন- প্রোগ্রামে printf() কমান্ডের পরিবর্তে print() লেখা। সিনটাক্স ভুলকে কম্পাইল টাইম ভুলও বলা হয়।

**লজিক্যাল বা যৌক্তিক ভুলঃ** প্রোগ্রামে যুক্তির ভুল থাকলে তাকে বলে লজিক্যাল ভুল। সাধারণত সমস্যা ঠিকমতো না বুঝার জন্যই এ ভুল হয়। যেমন-  $a > b$  এর স্থলে  $a < b$  বা  $s = a + b$  এর স্থানে  $s = a - b$  লিখলে লজিক্যাল ভুল হয়। লজিক্যাল ভুলের ক্ষেত্রে একটি উত্তর পাওয়া যায় যদিও তা ভুল। এক্ষেত্রে অনুবাদক প্রোগ্রাম কোনো ভুলের বার্তা ছাপায় না বলে লজিক্যাল ভুল সংশোধন করা খুব কঠিন।

**ডেটাগত ভুলঃ** যদি কোনো প্রোগ্রামে ডেটা দিতে ভুল হয় তখন তাকে ডেটাগত ভুল বলে।

**রান টাইম এরর -** কম্পিউটারের প্রোগ্রামে ভুল ডেটা থাকলে অথবা ডেটার ফরমেট ভুল থাকলে রান টাইম এরর বলে।

**Syntax error-** কোনো প্রোগ্রামের ব্যাকরণগত ভুলকে সিনটাক্স ভুল বলে।

#### অ্যালগোরিদমঃ

কোনো একটি নির্দিষ্ট সমস্যা ধাপে ধাপে সমাধানের জন্য যুক্তিসম্মত সসীম সংখ্যক পর্যায়ক্রমিক ধারা বর্ণনাকে একত্রে অ্যালগরিদম বলা হয়। কোনো সমস্যাকে কম্পিউটার প্রোগ্রামিং দ্বারা সমাধান করার পূর্বে কাগজে-কলমে সমাধান করার জন্যই অ্যালগরিদম ব্যবহার করা হয়। আরব গণিতবিদ ‘আল খারিজমী’ এর নাম অনুসারে অ্যালগরিদম নামকরণ করা হয়েছে। অ্যালগরিদমের প্রত্যেকটি ধাপসহ ইনপুট এবং আউটপুট স্পষ্টভাবে নির্ধারণ করতে হবে। সসীম সংখ্যক ধাপে সমস্যার সমাধান হতে হবে। অ্যালগোরিদমে কোন কম্পিউটার কোড থাকা যাবে না। বরং অ্যালগরিদম এমনভাবে লিখতে হবে যা একই ধরনের প্রোগ্রামিং ভাষার জন্য ব্যবহার করা যাবে।

#### অ্যালগরিদম তৈরির সুবিধাঃ

- ১। এটি একটি ধাপভিত্তিক উপস্থাপনা ফলে সহজে প্রোগ্রামের উদ্দেশ্য বোঝা যায়।
- ২। একটি অ্যালগরিদম একটি নির্দিষ্ট পদ্ধতি ব্যবহার করে।
- ৩। এটি কোনও প্রোগ্রামিং ভাষার উপর নির্ভরশীল নয়, তাই প্রোগ্রামিং জ্ঞান ছাড়াই যেকারো পক্ষে এটি বোঝা সহজ।
- ৪। একটি অ্যালগরিদমের প্রতিটি ধাপের নিজস্ব লজিক্যাল ক্রম আছে তাই এটি ডিবাগ করা সহজ।
- ৫। প্রোগ্রাম পরিবর্তন ও পরিবর্তনে সহায়তা করে।

#### #দুটি সংখ্যার গড় নির্ণয়ের অ্যালগরিদম-

- ধাপ-১: শুরু করি।
- ধাপ-২: ইনপুট হিসেবে a ও b চলকের মান গ্রহণ করি।
- ধাপ-৩:  $avg = (a+b)/2$  নির্ণয় করি।
- ধাপ-৪: ফলাফল হিসেবে avg চলকের মান প্রদর্শন করি।
- ধাপ-৫: শেষ করি।

#### ফ্লোচার্ট বা প্রবাহ চিত্রঃ

যে চিত্রভিত্তিক পদ্ধতিতে বিশেষ কতকগুলো চিহ্নের সাহায্যে কোনো একটি নির্দিষ্ট সমস্যার সমাধান করা হয় তাকে ফ্লোচার্ট বলা হয়। অন্যভাবে বলা যায়, অ্যালগরিদমের চিত্ররূপই হল ফ্লোচার্ট। ফ্লোচার্টের সাহায্যে প্রোগ্রাম বোঝা সহজ হয় বলে এটি প্রোগ্রামার ও ব্যবহারকারীর মাঝে সংযোগ রক্ষার জন্য ব্যবহৃত হয়।

#### ফ্লোচার্ট তৈরি করার নিয়মাবলিঃ

- ১। প্রতিটি ফ্লোচার্টের অবশ্যই একটি শুরু এবং শেষ থাকবে।

- ২। নিয়ন্ত্রণ প্রবাহ অবশ্যই টপ থেকে শুরু হবে।
- ৩। নিয়ন্ত্রণ প্রবাহ অবশ্যই বটম থেকে শেষ হবে।
- ৪। প্রচলিত চিহ্ন বা প্রতীক ব্যবহার করে ফ্লোচার্ট তৈরি করতে হবে।
- ৫। তীর চিহ্ন দিয়ে নিয়ন্ত্রণ প্রবাহ দেখাতে হবে।
- ৬। ফ্লোচার্টে কোন প্রোগ্রামিং ভাষা ব্যবহার করা যাবে না।
- ৭। চিহ্ন বা প্রতীক গুলো ছোট বড় হলে ক্ষতি নাই তবে আকৃতি ঠিক থাকতে হবে।

### ফ্লোচার্টের সুবিধাঃ

- ১। একটি প্রোগ্রামের যুক্তির মধ্যে যোগাযোগের চমৎকার উপায় হলো ফ্লোচার্ট।
- ২। ফ্লোচার্ট ব্যবহার করে সমস্যা বিশ্লেষণ করা সহজ।
- ৩। প্রোগ্রাম উন্নয়নের সময়, ফ্লোচার্ট একটি ব্লুপ্রিন্টের ভূমিকা পালন করে।
- ৪। ফ্লোচার্ট এর সাহায্যে প্রোগ্রাম বা সিস্টেম রক্ষণাবেক্ষণ সহজ হয়।
- ৫। ফ্লোচার্টকে যেকোনো প্রোগ্রামিং ভাষার কোডে রূপান্তর করা সহজ।

ফ্লোচার্টের প্রকারভেদঃ ফ্লোচার্টকে প্রধানত দুইভাগে ভাগ করা যায়। যথা-

### ১। সিস্টেম ফ্লোচার্ট






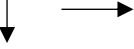
সিস্টেম ফ্লোচার্টে ডেটা গ্রহণ, প্রক্রিয়াকরণ, সংরক্ষণ এবং ফলাফল প্রদর্শনের প্রবাহ দেখানো হয়। কোন সিস্টেমের কার্যপ্রণালি বোঝাতে সিস্টেম ফ্লোচার্ট ব্যবহৃত হয়।

### ২। প্রোগ্রাম ফ্লোচার্ট

প্রোগ্রাম ফ্লোচার্টে প্রোগ্রামের বিভিন্ন ধাপের বিস্তারিত বিবরণ দেওয়া হয়। এছাড়া প্রোগ্রামের ভুল নির্ণয় ও সংশোধনে প্রোগ্রাম ফ্লোচার্ট ব্যবহৃত হয়। ফ্লোচার্টের চার ধরনের ঠিকাকার আছে-

- ১। সরল অনুক্রম (Simple Sequence)
- ২। নির্বাচন বা সিলেকশন (Selection)
- ৩। পুনরাবৃত্তি বা লুপ (Loop)
- ৪। জাম্প (Jump)

### ফ্লোচার্টে ব্যবহৃত প্রতীক সমূহ এবং এদের ব্যবহারঃ

প্রতীক	প্রতীকের নাম	ব্যবহার
	ডিম্বক (Oval)	শুরু এবং শেষ নির্দেশ করে।
	সামান্তরিক	ইনপুট ও আউটপুট নির্দেশ করে।
	আয়তাকার	প্রক্রিয়াকরণ নির্দেশ করে।
	হীরক	এই প্রতীকের মধ্যে শর্ত / সিদ্ধান্ত লেখা হয়।
	বৃত্ত/কানেক্টর	একাধিক ফলাফল সংযোগ করে।
	তীর চিহ্ন	প্রবাহের দিক নির্দেশে ব্যবহৃত হয়।

### সুডোকোড (Pseudo Code):

সুডো (Pseudo) একটি গ্রিক শব্দ যার অর্থ হচ্ছে ছদ্ম বা সত্য নয়। প্রোগ্রামিং করার পূর্বে মূলত সুডোকোড করা হয়। যা কমেন্ট আকারেও প্রোগ্রামে ব্যবহার করা হয়।

### সুডোকোড এর ব্যবহার

১. পরিকল্পনা: বড় কোনো প্রোগ্রাম লেখার আগে তার লজিক বা ধাপগুলো ঠিক করার জন্য।
২. যোগাযোগ: যারা প্রোগ্রামিং জানেন না, তাদের কাছে প্রোগ্রামের কর্মপদ্ধতি ব্যাখ্যা করার জন্য।
৩. ভুল কমানো: কোডিং শুরু করার আগেই লজিকে কোনো ভুল আছে কি না তা যাচাই করা যায়।

## MCQ

১. উৎস (Source) প্রোগ্রামকে বস্তু (Object) প্রোগ্রামে রূপান্তর করে যে প্রোগ্রাম তাকে বলে—

- A. লিঙ্কার B. লোডার  
C. ডিবাগার D. অনুবাদক প্রোগ্রাম

Ans: D

২. মেশিন ভাষায় লেখা প্রোগ্রামকে কী বলা হয়?

- A. সোর্স প্রোগ্রাম B. হাই লেভেল প্রোগ্রাম  
C. অবজেক্ট প্রোগ্রাম D. অ্যাসেম্বলি প্রোগ্রাম

Ans: C

৩. নিচের কোনটি অনুবাদক প্রোগ্রামের প্রকারভেদ?

- A. Debugger B. Editor  
C. Compiler D. Loader

Ans: C

৪. অ্যাসেম্বলার কোন ভাষাকে মেশিন ভাষায় রূপান্তর করে?

- A. উচ্চস্তরের ভাষা B. অ্যাসেম্বলি ভাষা  
C. মেশিন ভাষা D. প্রাকৃতিক ভাষা

Ans: B

৫. সিস্টেম ফ্লোচার্টে কী দেখানো হয়?

- A. শুধু কোড B. হার্ডওয়্যার নকশা  
C. সিস্টেমের কার্যপ্রণালি D. অ্যালগরিদম

Ans: C

৬. ইন্টারপ্রেটার কোনভাবে প্রোগ্রাম অনুবাদ করে?

- A. সম্পূর্ণ একসাথে B. লাইন বাই লাইন  
C. ব্লক আকারে D. মডিউল আকারে

Ans: B

৭. কম্পাইলার প্রোগ্রামের ভুল কিভাবে দেখায়?

- A. একবারে সব B. লাইন বাই লাইন  
C. দেখায় না D. রান টাইমে

Ans: A

৮. ডিবাগিং ও টেস্টিং কোনটির ক্ষেত্রে দূত?

- A. কম্পাইলার B. অ্যাসেম্বলার  
C. ইন্টারপ্রেটার D. লোডার

Ans: C

৯. প্রতিবার কাজের আগে অনুবাদ করতে হয়—

- A. কম্পাইলারে B. অ্যাসেম্বলারে  
C. ইন্টারপ্রেটারে D. লিঙ্কারে

Ans: C

১০. পূর্ণাঙ্গ যান্ত্রিক প্রোগ্রাম তৈরি হয়—

- A. ইন্টারপ্রেটারে B. কম্পাইলারে  
C. ডিবাগারে D. এডিটরে

Ans: B

১১. একটি পূর্ণাঙ্গ প্রোগ্রামের মৌলিক অংশ কয়টি?

- A. ২টি B. ৩টি C. ৪টি D. ৫টি

Ans: B

১২. প্রোগ্রাম সংগঠনের প্রথম ধাপ কোনটি?

- A. প্রসেস B. আউটপুট  
C. ইনপুট D. ডিবাগিং

Ans: C

১৩. আদর্শ প্রোগ্রামের একটি বৈশিষ্ট্য নয়—

- A. সহজ ও বোধগম্য B. বেশি মেমোরি ব্যবহার  
C. রক্ষণাবেক্ষণযোগ্য D. ডিবাগিং সহজ

Ans: B

১৪. প্রোগ্রাম তৈরির প্রথম ধাপ কোনটি?

- A. কোডিং B. সমস্যা বিশ্লেষণ  
C. সমস্যা নির্দিষ্টকরণ D. ডকুমেন্টেশন

Ans: C

১৫. প্রোগ্রামের ভুলকে কী বলা হয়?

- A. Error B. Debug C. Virus D. Bug

Ans: D

১৬. সিনটাক্স ভুলকে আর কী বলা হয়?

- A. লজিক্যাল ভুল B. রান টাইম ভুল  
C. কম্পাইল টাইম ভুল D. ডেটাগত ভুল

Ans: C

১৭. কোন ভুলে ভুল আউটপুট পাওয়া যায় কিন্তু Error Message আসে না?

- A. সিনটাক্স ভুল B. ডেটাগত ভুল  
C. লজিক্যাল ভুল D. রান টাইম ভুল

Ans: C

১৮. ভুল ডেটা বা ফরমেটের কারণে যে ভুল হয়—

- A. সিনটাক্স ভুল B. রান টাইম এরর  
C. লজিক্যাল ভুল D. কম্পাইল টাইম ভুল

Ans: B

১৯. ধাপে ধাপে সমস্যার সমাধানের লিখিত রূপকে বলে—

- A. ফ্লোচার্ট B. প্রোগ্রাম  
C. অ্যালগরিদম D. সুডোকোড

Ans: C

২০. অ্যালগরিদম নামকরণ করা হয়েছে—