



Programming Essentials

Subject Code: 66631



Mark Distribution (for 150 Marks)

Theory Marks		Practical Marks	
Midterm	20	PC	25
Class test	10	PF	25
Quiz test	10	-	-
Final	60	-	-
Total	100	Total	50

Outline of Chapter - 01



কম্পিউটার প্রোগ্রাম ও প্রোগ্রামিং
প্রোগ্রামিং এর ব্যাখ্যা ও প্রকারভেদ
প্রোগ্রামিং ভাষার প্রজন্মের বিবৃতি
অনুবাদক প্রোগ্রামের ব্যাখ্যা
অ্যালগরিদম ও ফ্লোচার্ট
সমস্যা ও সমাধান
প্রোগ্রাম পরিকল্পনা প্রক্রিয়া

□ প্রোগ্রাম কি?

যন্ত্রের মাধ্যমে কোন সমস্যা সমাধানের জন্য প্রোগ্রামিং ভাষা ব্যবহার করে প্রোগ্রামারের দেওয়া প্রয়োজনীয় নির্দেশের সমষ্টিকে প্রোগ্রাম বলা হয়।

□ প্রোগ্রামিং কি?

প্রোগ্রামিং ভাষা ব্যবহার করে কোন যন্ত্রকে নির্দেশনা দেওয়াকে বলা হয় প্রোগ্রামিং। অন্যভাবে বলা যায়, প্রোগ্রাম রচনার পদ্ধতি বা কৌশলকে প্রোগ্রামিং বলা হয়।

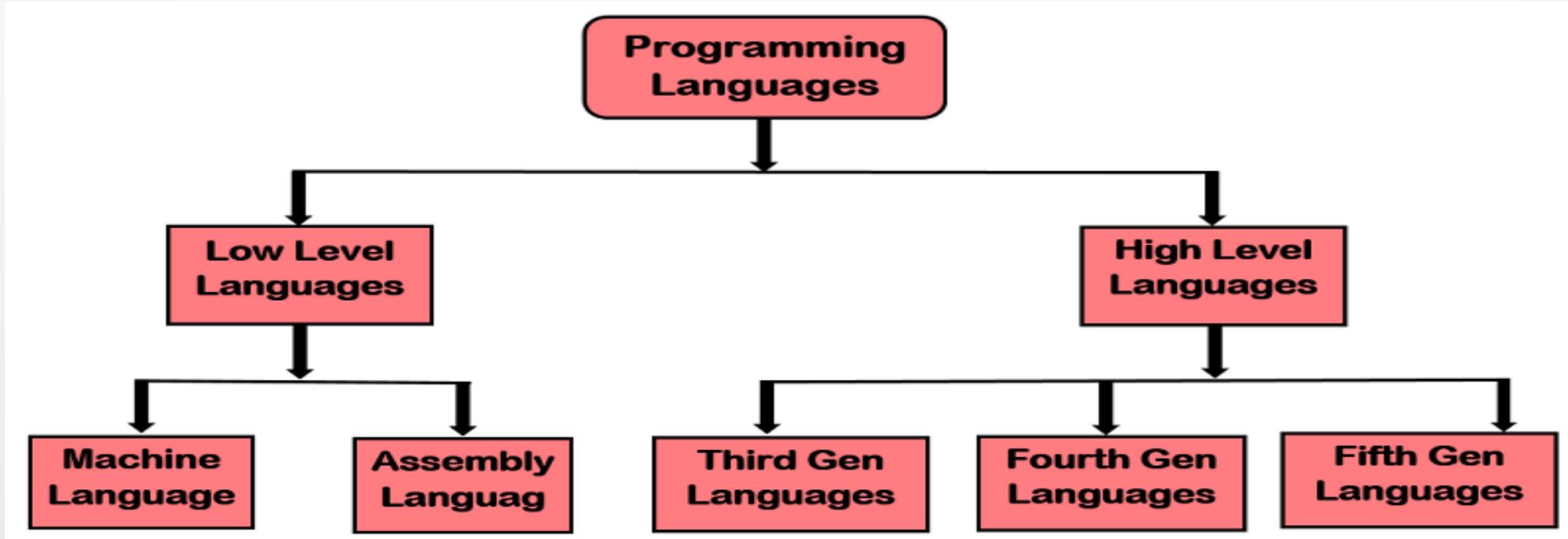
□ প্রোগ্রামিং ল্যাঙ্গুয়েজ কি?

যে ভাষার সাহায্যে প্রোগ্রাম তৈরি করা হয় তাকে প্রোগ্রামিং ল্যাঙ্গুয়েজ বলে।

কয়কটি জনপ্রিয় ল্যাঙ্গুয়েজ হচ্ছে C, C++, Java, Python, C# ইত্যাদি।

১.২ প্রোগ্রামিং এর ব্যাখ্যা ও প্রকারভেদ

- প্রোগ্রামিং ল্যাঙ্গুয়েজের প্রকারভেদ (Types of Programming Language)





□ মেশিন বা যান্ত্রিক ভাষা (Machine Language) কি?

যে ভাষায় শুধুমাত্র ০ এবং ১ ব্যবহার করে প্রোগ্রাম লেখা হয় তাকে মেশিন বা যান্ত্রিক ভাষা বলে। কম্পিউটারের নিজস্ব ভাষা হচ্ছে মেশিন ভাষা। এটি কম্পিউটারের মৌলিক ভাষা। এই ভাষায় শুধু মাত্র ০ এবং ১ ব্যবহার করা হয় বলে এই ভাষায় দেওয়া কোনো নির্দেশ কম্পিউটার সরাসরি বুঝতে পারে। এর সাহায্যে কম্পিউটারের সাথে সরাসরি যোগাযোগ করা যায়। এটি প্রথম প্রজন্মের এবং নিম্নস্তরের ভাষা। মেশিন ভাষায় লেখা প্রোগ্রামকে অবজেক্ট বা বস্তু প্রোগ্রাম বলা হয়

□ মেশিন ভাষার সুবিধা:

- ১। মেশিন ভাষার সবচেয়ে বড় সুবিধা হচ্ছে কম্পিউটারের সাথে সরাসরি যোগাযোগ করা যায়।
- ২। মেশিন ভাষায় লেখা প্রোগ্রাম নির্বাহের জন্য কোনো প্রকার অনুবাদক প্রোগ্রামের প্রয়োজন হয় না। ফলে দ্রুত কাজ করে।
- ৩। মেশিন ভাষায় লেখা প্রোগ্রামে অতি অল্প মেমোরি প্রয়োজন হয়।

□ মেশিন ভাষার অসুবিধা:

- ১। শুধু ০ ও ১ ব্যবহার করা হয় বলে মেশিন ভাষা শেখা কষ্টকর এবং এই ভাষা ব্যবহার করে প্রোগ্রাম লেখাও কষ্টসাধ্য।
- ২। এই ভাষায় লেখা কোনো প্রোগ্রাম সাধারণত বোঝা যায় না।
- ৩। এই ভাষায় প্রোগ্রাম লিখতে প্রচুর সময় লাগে এবং ভুল হবার সম্ভাবনা খুব বেশি থাকে। ভুল হলে তা বের করা এবং ভুল-ত্রুটি দূর করা অর্থাৎ ডিবাগিং কষ্টসাধ্য।
- ৪। এই ভাষার সবচেয়ে বড় অসুবিধা হচ্ছে এক ধরনের কম্পিউটারের জন্য লেখা প্রোগ্রাম অন্য ধরনের কম্পিউটারে ব্যবহার করা যায় না। অর্থাৎ যন্ত্র নির্ভর ভাষা।
- ৫। এই ভাষায় প্রোগ্রাম রচনার ক্ষেত্রে কম্পিউটারের অভ্যন্তরীণ সংগঠন ভালোভাবে জানতে হয়।

□ অ্যাসেম্বলি ভাষা (Assembly Language) কি ?

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 (mnemonic) এবং এটি সর্বোচ্চ পাঁচটি লেটারের সমন্বয়ে হয়, যেমন- SUB(বিয়োগের জন্য), MUL(গুণের জন্য), ADD(যোগের জন্য), DIV(ভাগের জন্য) ইত্যাদি। এই বৈশিষ্ট্যের জন্য এই ভাষাকে সাংকেতিক ভাষাও বলা হয়। অ্যাসেম্বলি ভাষা দ্বিতীয় প্রজন্মের এবং নিম্নস্তরের ভাষা। দ্বিতীয় প্রজন্মের কম্পিউটারে এই ভাষা ব্যাপকভাবে প্রচলিত ছিল। এই ভাষায় লেখা প্রোগ্রাম অনুবাদের প্রয়োজন হয় এবং অনুবাদক প্রোগ্রাম হিসেবে অ্যাসেম্বলার ব্যবহৃত হয়।



□ অ্যাসেম্বলি ভাষার সুবিধা:

- ১। অ্যাসেম্বলি 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। এই ভাষার সবচেয়ে বড় অসুবিধা হচ্ছে এক ধরনের কম্পিউটারের জন্য লেখা প্রোগ্রাম অন্য ধরনের 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 ভাষা (High Level Language) কি?

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 ব্যবহার:

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



মধ্যমস্তরের ভাষা (Mid Level Language) কি?

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।

১.৩ প্রোগ্রামিং ভাষার প্রজন্মের বিবৃতি

□ প্রোগ্রামিং ভাষার প্রজন্ম : 1945 থেকে শুরু করে এ পর্যন্ত যত প্রোগ্রামিং ভাষা আবিষ্কৃত হয়েছে তাদেরকে বৈশিষ্ট্য অনুযায়ী পাঁচটি প্রজন্মে ভাগ করা হয়েছে।

- **প্রথম প্রজন্ম** – First Generation(1945-1949)
 - Machine Language (যান্ত্রিক ভাষা)
- **দ্বিতীয় প্রজন্ম** – Second Generation(1950-1959)
 - Assembly Language (অ্যাসেম্বলি ভাষা)
- **তৃতীয় প্রজন্ম** –Third Generation(1960-1969)
 - High Level Language (উচ্চস্তরের ভাষা)
- **চতুর্থ প্রজন্ম** – Fourth Generation(1970-1979)
 - Very High Level Language (অতি উচ্চস্তরের ভাষা)
- **পঞ্চম প্রজন্ম** – Fifth Generation(1980-present)
 - Natural Language(স্বাভাবিক ভাষা)
- **ষষ্ঠ প্রজন্ম** – (6th Generation)
 - Artificial Language(কৃত্রিম বুদ্ধিমত্তা ভাষা)

□ অনুবাদক প্রোগ্রাম:

- 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। এটি অ্যাসেম্বলি ভাষায় লেখা প্রোগ্রাম বা নেমোনিক কোডকে যান্ত্রিক ভাষায় রূপান্তর করে। এক্ষেত্রে প্রোগ্রামে কোনো ভুল থাকলে Error Message দেয়।

□ প্রধান কাজ সমূহঃ

- ১। নেমোনিক কোডকে মেশিন ভাষায় অনুবাদ করা।
- ২। সাংকেতিক ঠিকানাকে মেশিন ভাষার ঠিকানায় রূপান্তর করা।
- ৩। সব নির্দেশ ও ডেটা প্রধান মেমোরিতে রাখা।
- ৪। প্রোগ্রামে কোনো ভুল থাকলে Error Message দেওয়া।
- ৫। প্রোগ্রামের সকল ভুল সংশোধনের পর প্রোগ্রাম কন্ট্রোলকে জানানো ইত্যাদি।

□ কম্পাইলারঃ

- কম্পাইলার হলো এক ধরনের অনুবাদক প্রোগ্রাম যা উচ্চস্তরের ভাষায় লেখা প্রোগ্রামকে মেশিন বা যান্ত্রিক ভাষায় রূপান্তর করে। অর্থাৎ উৎস প্রোগ্রামকে বস্তু প্রোগ্রামে রূপান্তর করে। কম্পাইলার দুই ধাপে অনুবাদকের কাজ সম্পন্ন করে –
- প্রথম ধাপে কম্পাইলার উৎস প্রোগ্রামটি পড়ে এবং বস্তু প্রোগ্রামে রূপান্তর করে। এই ধাপে, সোর্স প্রোগ্রামে যদি কোন ভুল থাকে, তবে তা সংশোধন করার জন্য কম্পাইলার ব্যবহারকারীকে Error Message দেয়। এই Error Message কে কম্পাইলড টাইম ডায়াগনোস্টিক Error Message বলে। একবার প্রোগ্রাম কম্পাইল হয়ে গেলে পরবর্তীতে আর কম্পাইল করার প্রয়োজন হয় না। দ্বিতীয় ধাপে উপাত্ত বা ডেটার ভিত্তিতে ফলাফল প্রদর্শনের জন্য বস্তু প্রোগ্রামকে নির্বাহ করানো হয়।

□ কম্পাইলারের কাজঃ

- ১। উৎস প্রোগ্রামের স্টেটমেন্ট সমূহকে বস্তু প্রোগ্রামে বা মেশিন ভাষায় রূপান্তর।
- ২। সংশ্লিষ্ট সাব-রুটিন এর সাথে সংযোগের ব্যবস্থা প্রদান।
- ৩। প্রধান মেমোরির পরিসর চিহ্নিতকরণ।
- ৪। প্রোগ্রাম ভুল থাকলে অনুবাদের সময় ভুলের তালিকা প্রণয়ন।



□ **কম্পাইলারের সুবিধা:**

- ১। কম্পাইলার সম্পূর্ণ প্রোগ্রামটিকে একসাথে অনুবাদ করে। ফলে প্রোগ্রাম নির্বাহ দ্রুত হয়।
- ২। কম্পাইলারের মাধ্যমে রূপান্তরিত প্রোগ্রাম সম্পূর্ণরূপে মেশিন ভাষায় রূপান্তরিত হয়।
- ৩। একবার প্রোগ্রাম কম্পাইল করা হলে পরবর্তিতে আর কম্পাইলের প্রয়োজন হয় না।

□ **কম্পাইলারের অসুবিধা:**

- ১। কম্পাইলার প্রোগ্রামের সবগুলো ভুল একসাথে প্রদর্শন করে ফলে প্রোগ্রাম সংশোধনে বেশি সময় লাগে।
- ২। প্রোগ্রাম ডিবাগিং ও টেস্টিং এর কাজ ধীরগতি সম্পন্ন।
- ৩। কম্পাইলার বড় ধরনের প্রোগ্রাম হওয়ায় ইহা সংরক্ষণে মেমোরিতে বেশি জায়গা প্রয়োজন।

□ ইন্টারপ্রেটার:

ইন্টারপ্রেটারও কম্পাইলারের মতো এক ধরনের অনুবাদক প্রোগ্রাম যা উচ্চস্তরের ভাষায় লেখা প্রোগ্রামকে লাইন বা লাইন মেশিন বা যান্ত্রিক ভাষায় রূপান্তর করে। এক্ষেত্রে কম্পাইলারের সাথে পার্থক্য হল, কম্পাইলার সম্পূর্ণ সোর্স প্রোগ্রামকে একসাথে অবজেক্ট প্রোগ্রামে রূপান্তর করে এবং সর্বশেষ ফলাফল প্রদান করে কিন্তু ইন্টারপ্রেটার সোর্স প্রোগ্রামটিকে লাইন-বাই-লাইন অবজেক্ট প্রোগ্রামে রূপান্তর করে এবং তাৎক্ষণিক ফলাফল প্রদর্শন করে।

□ ইন্টারপ্রেটারের কাজ:

- ১। উৎস প্রোগ্রামের স্টেটমেন্ট সমূহকে বস্তু প্রোগ্রামে বা মেশিন ভাষায় রূপান্তর।
- ২। সংশ্লিষ্ট সাব-রুটিন এর সাথে সংযোগের ব্যবস্থা প্রদান।
- ৩। প্রধান মেমোরির পরিসর চিহ্নিতকরণ।
- ৪। প্রোগ্রাম ভুল থাকলে অনুবাদের সময় ভুলের তালিকা প্রণয়ন।

□ ইন্টারপ্রেটারের সুবিধা:

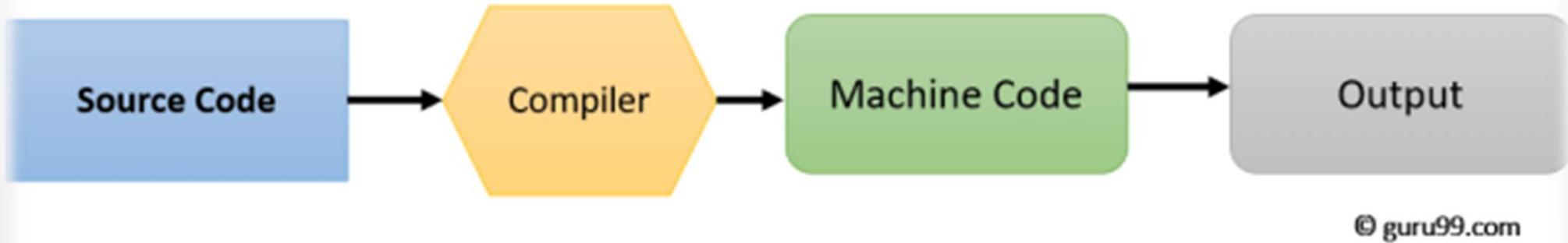
- ১। ইন্টারপ্রেটার এর সবচেয়ে বড় সুবিধা হল এটি ইউজার ফ্রেন্ডলি।
- ২। এটি ব্যবহারে প্রোগ্রামের ভুল সংশোধন করা এবং পরিবর্তন করা সহজ হয়।
- ২। ইন্টারপ্রেটার প্রোগ্রাম আকারে ছোট হয় বলে মেমোরিতে কম জায়গা দখল করে।
- ৩। এটি সাধারণত ছোট কম্পিউটারে ব্যবহার করা হয়।

□ ইন্টারপ্রেটারের অসুবিধা:

- ১। ইন্টারপ্রেটার যেহেতু প্রোগ্রাম লাইন-বাই-লাইন অনুবাদ করে, তাই অনুবাদ করতে কম্পাইলারের তুলনায় বেশি সময় প্রয়োজন।
- ২। ইন্টারপ্রেটার এর মাধ্যমে রূপান্তরিত প্রোগ্রাম সম্পূর্ণরূপে মেশিন প্রোগ্রামে রূপান্তরিত হয় না।
- ৩। প্রত্যেকবার প্রোগ্রাম নির্বাহের সময় অনুবাদ করার প্রয়োজন হয়।



How Compiler Works



How Interpreter Works



কম্পাইলার এবং ইন্টারপ্রেটারের মধ্যে পার্থক্য:

কম্পাইলার	ইন্টারপ্রেটার
কম্পাইলার একটি অনুবাদক প্রোগ্রাম যা উচ্চস্তরের প্রোগ্রামিং ভাষায় লেখা একটি সম্পূর্ণ প্রোগ্রামকে একসাথে অনুবাদ করে।	ইন্টারপ্রেটারও এক ধরনের অনুবাদক প্রোগ্রাম যা উচ্চস্তরের প্রোগ্রামিং ভাষায় লেখা একটি প্রোগ্রামকে লাইন বাই লাইন অনুবাদ করে।
ফলে প্রোগ্রাম নির্বাহের জন্য কম সময় প্রয়োজন।	ফলে প্রোগ্রাম নির্বাহের জন্য বেশি সময় প্রয়োজন।
কম্পাইলার দ্বারা একটি প্রোগ্রাম একবার অনুবাদ করা হলে প্রতিবার কাজের পূর্বে পুনরায় অনুবাদ করার প্রয়োজন হয় না।	ইন্টারপ্রেটার দ্বারা একটি প্রোগ্রাম অনুবাদ করা হলে প্রত্যেকবার প্রোগ্রাম নির্বাহের সময় অনুবাদ করার প্রয়োজন হয়।
কম্পাইলার দ্বারা একটি প্রোগ্রাম অনুবাদ করলে পূর্ণাঙ্গ যান্ত্রিক প্রোগ্রামে রূপান্তরিত হয়।	ইন্টারপ্রেটার দ্বারা একটি প্রোগ্রাম অনুবাদ করলে পূর্ণাঙ্গ যান্ত্রিক প্রোগ্রামে রূপান্তরিত হয় না।
ডিবাগিং এবং টেস্টিং এর ক্ষেত্রে ধীর গতি সম্পন্ন।	ডিবাগিং ও টেস্টিং এর ক্ষেত্রে দ্রুত গতি সম্পন্ন।



Thanks To All