

Introduction to Programming

Basic Model of Computation

Computation Model में input device का प्रयोग करके CPU को instruction दिये जाते हैं जो CPU के द्वारा Primary Memory में transfer कर दिये जाते हैं ताकि इन्हें temporary रूप से स्टोर किया जा सके।

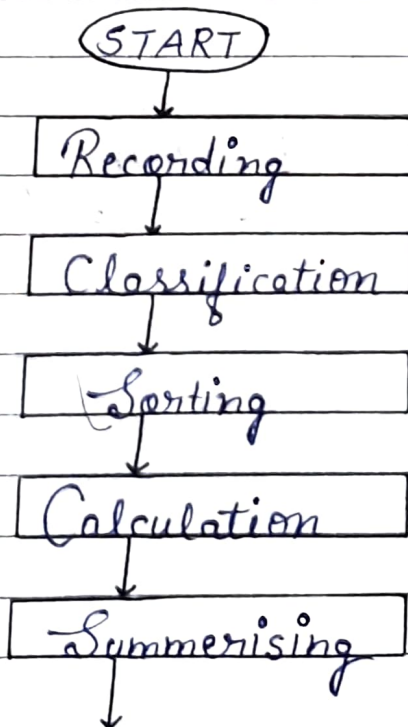
CPU डेटा की एक कॉपी Calculation के लिए ALU के पास भेजता है।

यहाँ पर Mathematical & logical Calculation की जाती है।

तथा इसके बाद डेटा Output में convert करके Output device के द्वारा Show कर दिया जाता है।

Basic data processing Model -

Processing Model में 6 Model Steps होते हैं -



Reporting

↓
STOP

1. Recording — Input device के द्वारा सर्वप्रथम डेटा input किया जाता है या record Memory में स्टोर कर दिया जाता है।
2. Classification — डेटा अलग-अलग record करने के बाद इसे class के रूप में वर्गीकृत कर दिया जाता है।
3. Sorting — डेटा को arrange करने के लिए Sorting की जाती है।
4. Calculation — Classified or Sorted data पर Computer के द्वारा process की जाती है। तथा आवश्यकतानुसार result create किया जाता है।
5. Summarization — Process के समय कुछ extra result create होते जाते हैं। इस स्टेप में Memory से extra result को हटाया जाता है तथा आवश्यक result ही रखे जाते हैं।
6. Reporting — इस Step में प्राप्त रिजल्ट को स्क्रीन पर दिखाया जाता है।

Programming Language -

Programming Language कम्प्यूटर तथा User के बीच Communication का एक माध्यम है।

प्रोग्रामिंग language के द्वारा computer को विभिन्न कार्य करने के लिए instruction दिये जाते हैं।

विभिन्न प्रोग्रामिंग language विकसित की जा चुकी हैं।

Machine language -

यह सबसे पुरानी language है। जिसका प्रयोग first generation के प्रोग्राम को लिखने के लिए किया गया था। इसमें 0 तथा 1 दो नम्बर का प्रयोग किया जाता है जिस कारण इसे Binary language कहा जाता है। मशीन पर निर्भर होने के कारण एक कम्प्यूटर के लिए बनाया गया प्रोग्राम दूसरे कम्प्यूटर पर कार्य नहीं करता।

लाभ :- इसमें translator की आवश्यकता नहीं होती। Computer direct इस भाषा को समझता है। इस भाषा में बनाये गये प्रोग्राम तेज गति से कार्य करते language हैं।

हानि :- इस language को सीखना, प्रोग्रामिंग करना काफी hard बड़े होते हैं। इसमें बनाये गये प्रोग्राम

Assembly Language - यह second generation की language है। इस language में प्रोग्राम बनाने के लिए symbols & (0,1) बाइनरी नम्बर का प्रयोग किया गया।

लाभ: - इस language को सीखना तथा करना मशीन language से आसान होता है। Coding इसके प्रोग्राम मशीन language से छोटे होते हैं।

हानि: - यह भाषा मशीन पर निर्भर रहती है। एक कंप्यूटर के लिए बनाया गया प्रोग्राम दूसरे कंप्यूटर पर कार्य नहीं करता। यह language direct computer के द्वारा नहीं समझी जाती। जिसके लिए इसमें translator की आवश्यकता होती है।

High level language (HLL) -

यह Third generation की language है। जिसमें प्रोग्राम बनाने की जगह binary no. का प्रयोग english के Simple words & Symbols किया जाता है।

यह procedural languages है।

Ex:- BASIC, COBOL, FORTRAN (First HLL), PASCAL, 'C', C++, PYTHON, JAVA etc.

लाभ:- यह language सीखना तथा कोडिंग करना आसान है। यह language मशीन या कंप्यूटर पर निर्भर नहीं करती। इसमें बनाये गये प्रोग्राम छोटे होते हैं।

हानि:- इसमें translator की आवश्यकता होती है।

Fourth Generation language -

इसे 4GL के नाम से जाना जाता है। इसमें छोटे छोटे प्रोग्राम बनाये जाते हैं जो कंप्यूटर को कमांड देने के लिए बनाये जाते हैं।

जैसे - किस कार्य को करना है।

Fifth Generation Language —

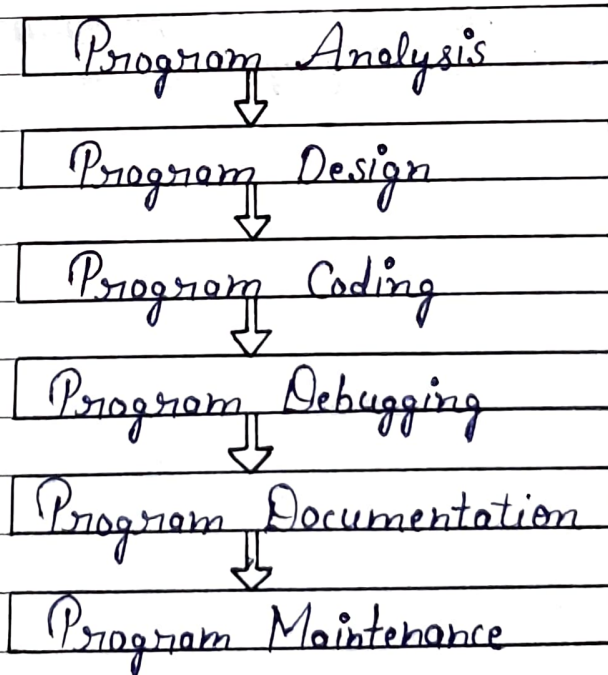
यह AI के लिए प्रसिद्ध है। ज्यादातर इसका प्रयोग robots बनाने के लिए किया जाता है। इस language 5GL या AI कहलाती है। generation

जैसे- LISP

Steps Of Programming —

कम्प्यूटर language में लिखे गये निर्देशों का व्यवस्थित सैट जो Computer किया जाता है, प्रोग्राम कहलाता है। प्रोग्रामर design के द्वारा प्रोग्रामिंग लिखने की प्रक्रिया प्रोग्रामिंग है।
 प्रोग्राम

Programming के 6: Step होते हैं -



Program Analysis -

इस स्टेप में दी गई Problem को analysis किया जाता है तथा यह पहचाना जाता है कि user से क्या input प्राप्त किया जायेगा जिस output का प्रयोग किया जायेगा आदि। Step प्रोग्राम की full information होनी चाहिए।

Program Design -

इस स्टेप में प्रोग्रामर प्रोग्राम की संरचना बनाता है। How chart के द्वारा किसी भी प्रोग्राम की Graphical presentation तैयार की जाती है।

Program Coding -

किसी भी प्रोग्रामिंग language का प्रयोग करके कम्प्यूटर पर प्रोग्राम लिखा जाता है। ये प्रोग्राम Coding होगा।

Program Debugging -

प्रोग्राम कोडिंग करने के बाद debugging के द्वारा प्रोग्राम में error check किये जाते हैं तथा इन्हें ठीक किया जाता है।

Program Documentation -

प्रोग्राम बनाने के साथ इस प्रोग्राम से रिलेटेड instruction manual या guide बनायी जाती है।

Program Maintenance —

User की आवश्यकतानुसार प्रोग्राम में संशोधन (correction) किये जाते हैं या इसे अपग्रेड किया जाता है। क्योंकि User की आवश्यकता समय के साथ बदलती रहती है।

Compilation —

Computer Program HLL में लिखे जाते हैं तथा Computer binary language (0,1) को समझता है।

प्रोग्राम को कंप्यूटर पर चलाने के लिए इसे binary language में convert करना पड़ता है। HLL को LL / Machine language कहलाती है।

HLL में लिखे गये सांख्यिक कोड कहलाते हैं तथा मशीनी भाषा या Codes में लिखे गये कोड कहलाते हैं।
Object Code binary language है।

Source Code → Compiler → Object Code

NOTE: Debugging के लिए use किये जाने वाले सॉफ्टवेयर Diagnostic Software कहलाते हैं।

Algorithm —

किसी भी प्रोग्राम को लिखना easy language में Step-by-Step है। इसे Algorithm कहलाता है।
सकता Program development का first step कहा जा

Advantage :-

- Algorithm को english language में लिखा जाता है जो english language पोग्रामर तथा आम व्यक्ति आसानी से समझ सकते हैं।
- Algorithm के द्वारा नया पोग्राम लिखना आसान हो जाता है।
- यह maintenance में मदद करता है।

Disadvantage :-

- एक प्रोब्लम के लिए दो व्यक्तियों के द्वारा अलग-अलग Algorithm लिखे जा सकते हैं तथा दोनों सही भी हो सकते हैं।
- Algorithm Process Manual Process है। इसलिए इसमें कंप्यूटर से कोई मदद नहीं मिलती।

Examples

Algorithm 1 - Principal, rate & time input करके Simple interest calculate करने का Algorithm लिखें।

Step 1 → Start

Step 2 → Read p, r, t

Step 3 → $si = (p \times r \times t) / 100$

Step 4 → Print si

Step 5 → Stop

p = int(input("Enter principal: "))

r = int(input("Enter rate: "))

t = int(input("Enter time: "))

si = (p * r * t) / 100

print("S.I. is", si)

Algorithm 2 - दो वैरिएबल A & B के मानों को बदलने के लिए एक Algorithm लिखें।

- Step 1 → Start
- Step 2 → Read A, B
- Step 3 → C = A
- Step 4 → A = B
- Step 5 → B = C
- Step 6 → Print A, B
- Step 7 → stop

C = 10
A = 20
B = 10

Read A, B
A, B = B, A

Algorithm 3 - तीन विषय के अंकों को input करने का Algorithm लिखें। तथा इनके योग और औसत और Calculate करें। प्रत्येक विषय percentage में है।
max. no. 50

- Step 1 → Start
- Step 2 → Read A, B, C
- Step 3 → Add = (A+B+C) add = (h+e+m)
- Step 4 → Aver. = (A+B+C)/3 average = (add)/3
- Step 5 → Per. = (A+B+C) X 100 / 150 perc. = (add X 100) / 150
- Step 6 → Print (add, average, Percentage) print (add, average, percentage)
- Step 7 → Stop

Algorithm 4 - तीन नंबर इनपुट करने के लिए Algorithm लिखें और उसमें सबसे बड़ी संख्या बात करें।

- Step 1 → Start
- Step 2 → Read A, B, C
- Step 3 → if A > B and A > C Then
- Step 4 → Print largest No. = A
- Step 5 → if B > A and B > C then
- Step 6 → Print largest No. = B

Step 7 → if $C > A$ and $C > B$ then

Step 8 → Print Largest No. = C

Step 9 → Stop

Algorithm 5

प्रथम 50 सम संख्याओं को प्रिंट करने के लिए एक एल्गोरिथ्म लिखें।

Step 1 → Start

Step 2 → Set $E = 2$

Step 3 → Repeat Step 4 & 5 while ($E \leq 100$)

Step 4 → Print E

Step 5 → $E = E + 2$

Step 6 → Stop

First 50 even no. (2 - 100)

$E = 2$ (first even no.)

$E \leq 100$

Print E, $E = 2$

$E = E + 2$

$E = 2 + 2$

$E = 4$

Print E, $E = 4$

$E = E + 2$

$E = 4 + 2$

$E = 6$

Algorithm 6

किसी वस्तु की बिक्री की मात्रा और इकाई मूल्य के लिए एक Algorithm लिखें। ग्राँस बिल राशि, टूट और नेट बिल राशि की गणना करें और प्रिंट करें। इसके लिए शर्तें निम्न प्रकार हैं।

Quantity Sold	Discount rate
Upto 150	13%
151 - 300	21%
More than 300	40%

- Step 1 → Start
- Step 2 → Read qty, Up int (input ("qty * Up"))
- Step 3 → gb = qty * Up
- Step 4 → if qty ≤ 150 then
- Step 5 → dis = (gb * 13) / 100
- Step 6 → if qty ≥ 151 and qty ≤ 300 then
- Step 7 → dis = (gb * 21) / 100
- Step 8 → if qty > 300 then
- Step 9 → dis = (gb * 40) / 100
- Step 10 → nb = gb - dis
- Step 11 → print gb, dis, nb
- Step 12 → Stop

Algorithm 7 प्रथम 50 प्राकृतिक संख्या प्रिंट करने के लिए लिखें। Algorithm

- Step 1 → Start
- Step 2 → Set N = 1, ~~S = 1~~
- Step 3 → Repeat Step 4 & 5 while (N ≤ 50)
- Step 4 → ~~S = S + N~~ print N
- Step 5 → ~~N = N + 1~~ N = N + 1
- Step 6 → Print S = Stop

0 + 1 = 1
1 + 1 = 2
2

Algorithm 8 A - B तक सभी जोड़ प्रिंट करने का Algorithm

- Step 1 → Start
- Step 2 → ~~Read A, B~~ S = 10, a = 0
- Step 3 → Repeat Step 4 & 5 while (S ≤ 20)
- Step 4 → a = a + S
- Step 5 → S = S + 1
- Step 6 → print a
- Step 7 → Stop

= Assignment Oper.
== Equality Oper.

Date: _____
Page: 12

Algorithm 9 एक नंबर इनपुट करने के लिए लिखें और
check करें कि यह प्राइम नं. है या
नहीं।

Step 1 → Start
Step 2 → Read n
Step 3 → $s = 2$
Step 4 → Repeat Step 5 to 11 while ($s \leq n-1$)
Step 5 → $mod = n \% s$
Step 6 → if $mod == 0$
Step 7 → print ("this no. is not prime")
Step 8 → break
Step 9 → $s = s + 1$
Step 10 → if $s == n$
Step 11 → print ("this is prime no.")
Step 12 → Stop

Algorithm 10 15 शब्दों तक की
लिखें एक Fibonacci Series
Algorithm लिखें।

Fibonacci Series = 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144,
233, 377

Step 1 → Start
Step 2 → Set $t = 1$, Set $f = -1$, $l = 1$
Step 3 → Repeat Step 4 to 7 while ($t \leq 15$)
Step 4 → $s = f + l$
Step 5 → print s
Step 6 → $f = l$
Step 7 → $l = s$
Step 8 → $t = t + 1$
Step 9 → Stop

Algorithm 11 एक नंबर इनपुट करके, इसकी तथ्यात्मक गणना (factorial) करने और प्रिंट करने के लिए एक Algorithm लिखें।

Step 1 → Start

Step 2 → Read n , set $fact = 1$

Step 3 → Repeat Step 4 & 5 while ($n \geq 1$)

Step 4 → $fact = fact \times n$

Step 5 → $n = n - 1$

Step 6 → Print $fact$

Step 7 → Stop

- Write a program to print your name.

```
n = input("Enter your name")  
print("Your name is", n)
```

- WAP to input radius & Calculate area of circle.

```
r = int(input("Enter radius of circle"))  
area = 3.147 * r * r  
print("area of circle is", area)
```