इन्टरनेट एंड वेब पेज डिजाइनिंग

Web Publishing

July 2, 2016 6,313 Views 25 Min Read

Web Publishing

कोई वेबसाइट तैयार करने और उसके पूरा हो जाने पर किसी होस्ट सर्वर पर अपलोड करने की प्रक्रिया को वेब पब्लिशिंग कहा जाता है।

वेब पब्लिशिंग के चरण-

कोई वेबसाइट तैयार करके उसे इंटरनेट पर सबके उपयोग के लिये डालने की प्रक्रिया एक लम्बी प्रक्रिया है, जो किसी पुस्तक के प्रकाशन की तरह कई चरणो मे पूरी की जाती है। इस पूरी प्रक्रिया को हम निम्नलिखित पॉच चरणो मे बॉट सकते है-

1. डोमेन नाम का रजिस्ट्रेशन करना।
2. वेब होस्टिंग।
3. वेबसाइट डिजाइन और विकास।
4. प्रचार या प्रमोशन।
5. रखरखाव।

किसी सफल वेबसाइट के लिये ये सभी चरण महत्वपूर्ण है।

1. डोमेन नेम रजिस्ट्रेशन

किसी वेबसाइट का डोमेन नाम उसका ऐसा नाम या पता है जिससे इंटरनेट के उपयोगकर्ता द्वारा उसको पहचाना और देखा जाता है। वेब प्रकाशन का पहला चरण उस वेबसाइट के लिये कोई डोमेन नाम तय करना और किसी वेब सर्वर पर उस डोमेन नाम को पंजीकृत करना होता है।

डोमेन नेम चुनना

अपनी वेबसाइट के लिये डोमेन नाम का चयन आपको स्वयं करना होता है। यह नाम ऐसा होना चाहिये जो पहले से किसी अन्य वेबसाइट का ना हो और आपकी साइट के उददेश्य तथा सामाग्री का संकेत देता हो। सामान्यतः कोई कंपनी अपने नाम जैसा ही डोमेन नाम पंजीकृत कराता है यदि वह उपलब्ध हो। यदि वह उपलब्ध नही है तो उससे मिलता जुलता कोई नाम लिया जा सकता है, जो उपलब्ध हो।

वेबसाइट के लिये डोमेन नाम चुनते समय निम्नलिखित बातो को ध्यान मे रखना चाहिए-

1. डोमेन नेम ऐसा होना चाहिए जो आपका या आपकी कंपनी का या आपके ब्रांड का नाम हो। ऐसा करने से आपके ग्राहको को आपकी वेबसाइट तक पहुचना सरल होगा, क्योकि किसी भी व्यक्ति के लिये भिन्न नामो को याद रखना संभव नही होता है।

2. डोमेन नाम जितना छोटा हो, उतना ही अच्छा रहता है। छोटे नामो को याद रखना या टाइप करना भी सरल होता है। डोमेन नाम 67 अक्षरो जितने लंबे भी हो सकते है।

3. डोमेन नेम मे हाइफन (-) या डैश (-) का उपयोग सामान्यतया: नही होना चाहिए।

4. डोमेन नेम मे कैपीटल अक्षरो के प्रयोग से बचना चाहिए। सामान्यतः सभी डोमेन नाम छोटे अक्षरो मे रखे जाते है।

5. डोमेन नेम मे बहुवचन का प्रयोग नही होना चाहिए, क्योकि कई बार उपयोगकर्ता बहुवचन अर्थात अक्षर टाईप करना भूल जाते है। इससे गलत वेबसाइट खुल जाती है।

6. डोमेन नेम मे आपकी साइट के उददेश्य का भी पता चलता है। अतः यदि आपकी साइट परोपकारी या स्वयंसेवी संस्था है, तो आपको कभी भी .COM नाम नही लेना चाहिए। इसके बजाह ण्वतह वाला डोमेन नाम लेना चाहिए।
यदि आप इन बातो का ध्यान मे रखते हुए अपना डोमेन नाम तय करेगे तो आपकी वेबसाइट के सफल होने की अधिक संभावना होगी।

डोमेन नाम पंजीकृत कराना

डोमेन नेम रजिस्ट्रि जिसे नेटवर्क इनफॉमेशन सेटंर जाता है, डोमेन नाम सिस्टम का एक भाग है जो डोमेन नामो को IP Address मे बदलता है। यह एक संगठन है जो डोमेन नाम के पंजीकरण का प्रबंध करता है।डोमेन नाम आवंटित करने की नीति बनाता है और उच्च स्तरीय डोमेन को आॅपरेट करता है। यह डोमेन नाम के रजिस्ट्रारो से अलग होता है। डोमेन नामो को रजिस्टर करने का कार्य Internet Assigned Numbers Authorty या (IANA) नामक समिति द्वारा किया जाता है। यह समिति उच्च स्तरीय डोमेन नामो को स्वयं पंजीकृत करती है और शेष कार्य को विभिन्न संगठनो पर छोड देती है।

विभिन्न देशो के डोमेन नाम रजिस्टर करने वाले अनेक संगठन और वेबसाइट है। आप उनमे से किसी भी निर्धारित शुल्क देकर अपना डोमेन नाम पंजीकृत करा सकते है। उदाहरण के लिये भारत मे Yaho.com, GoDaddy.com, Candidinfo.com, Sify.com, Dotster.com, Register.com आदि अनेक वेबसाइटे यह कार्य करती है। उनका शुल्क अलग अलग होता है। डोमेन नाम पंजीकरण सामान्यतः एक साल के लिये किया जाता है। यह अवधि समाप्त होने से पहले ही आप फिर शुल्क देकर पंजीकरण बढा सकते है। वैसे आप एक साथ कई साल का शुल्क जमा करके भी अपना डोमेन नाम कई वर्ष के लिये पंजीकृत करा सकते है। कई वेबसाइट आपको निशुल्क डोमेन नाम देने का भी प्रस्ताव रखती है, परन्तु ऐसा करने से पहले आपको अच्छी तरह विचार कर लेना चाहिए।

2. वेब होस्टिंग

यह वेब पब्लिशिंग का सबसे महत्वपूर्ण भाग कहा जाता है। यह अपने आॅफिस या दुकान के लिये कोई जगह किराये पर लेने के समान है। जब आप अपनी वेबसाइट के डोमेन नाम का पंजीकरण करा लेते है और उसको डिजाइन कर लेते है, तो उसको किसी वेब सर्वर पर स्टोर करना ही वेब होस्टिंग कहा जाता है। यह कार्य किसी अच्छे वेब सर्वर पर ही किया जाता है, क्योंकि वे आपकी वेबसाइट को 24 घंटे सक्रिय रखते है और उसका उपयोग इंटरनेट पर सभी को उपलब्ध कराते है।

3. वेबसाइट डिजाइन और विकास

कोई वेबसाइट किसी विशेष विषय पर सूचनाओ का एक संग्रह होती है। किसी वेबसाइट को डिजाइन करने से हमारा तात्पर्य उसके वेब पेजो का निर्माण और उन्हे किसी विशेष रूप मे व्यवस्थित करना होता है। विभिन्न वेब पेजो से मिलकर ही कोई वेबसाइट बनती है। किसी वेब पेजे मे उन मे उन सूचनाओ कोई भाग होता है जिनके लिये वेबसाइट को बनाया गया है। इस प्रकार आप किसी वेबसाइट को एक पुस्तक के रूप मे देख सकते है, जिसके प्रत्येक पृष्ठ को एक वेब पेज माना जा सकता है।

4. प्रचार या प्रमोशन

किसी वेबसाइट का प्रचार निम्नलिखित विधियो से किया जाता है-

1. समाचार पत्रो या पत्रिकाओ मे विज्ञापन द्वारा।
2. रेडियो, टीवी. आदि पर विज्ञापन।
3. अन्य प्रसिध्द वेबसाइटो पर विज्ञापन देना।
4. अपनी साइट को सर्च इंजनो जैसे गूगल और याहू के साथ जोडना।

5. रखरखाव

किसी वेबसाइट की सफलता के लिये यह आवश्यक है कि उसमे नवीनता और रोचकता रहे ताकि उसके आगंतुक उसमे आते रहे। इसके लिये आपको अपनी वेबसाइट निरन्तर सुधारने और पुरानी जानकारी हटाकर नई जानकारियॉ डालते रहने की आवश्यकता होती है। इसलिये अपनी वेबसाइट पर स्वयं भ्रमण करते रहना चाहिए तथा परिवर्तनो की आवश्यकता का पता लगाते रहना चाहिए। विशेष तौर पर आपके उत्पादो, सेवाओ और उनके मूल्यो की जानकारी नवीनतम होनी चाहिए।

ये भी जरुर देखें

- www क्या हैं?
- Internet Terminology
- Set background color in HTML table
- How to Read Newspaper Online

Share this:

More

Like this:

Like

Be the first to like this.

Tags syntax of domain name web publishing web publishing in hindi what is web publishing what is web publishing in hindi

Google Cloud Platform

Build scalable
Machine Learning models

TRY IT FREE

C Language

- What is Programming?
- C - Introduction
- C - Tokens, Identifiers & Keywords
- C - Data Types
- C - Variables
- C - First Program
- C - Flow Chart
- C - Operators
- C - Decision Making
- C - Looping
- C - Arrays
- C - Strings
- C - Functions
- C - Structures
- C - Unions
- C - Recursion
- C - Pointers
- C - Type Casting
- C - Storage Classes
- C - Console I/O
- C - File Handling
- C - Error Handling
- C - Header Files
- C - Preprocessor Directives

C in Hindi

- Introduction to **C in Hindi**
- Characteristics & Features of *C in Hindi*
- Limitations of C in Hindi
- Applications of C in Hindi
- A simple C program

Introduction to C language

C एक बहुत ही popular programming language है। इसे Middle level language माना जाता है। आज भी C की popularity में कोई कमी नहीं आई है। किसी भी दूसरी language को सिखने से पहले C को सीखना बहुत जरुरी है। क्योंकि C के बाद जितनी भी **programming languages** आई उन्होंने किसी न किसी तरह C के features को adopt किया है। आइये C के बारे में detail से जानने का प्रयास करते है।

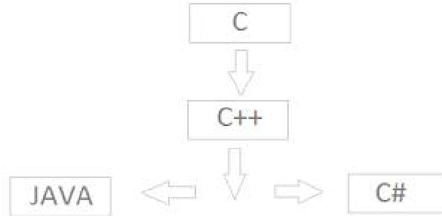
History

C एक **structured** general purpose programming language है। इसे **UNIX operating system** को दुबारा लिखने के लिए develop किया गया था। इससे पहले UNIX operating system को B language में लिखा गया था। पूरा UNIX operating system और उस पर run होने वाले ज्यादातर programs C language में ही लिखे गए है।

B language Ken Thompson के द्वारा 1970 में Bell laboratories में लिखी गई थी। B एक type less language थी। C language को **Dennis Ritchie** ने 1972 में **Bell laboratories** में ही develop किया था। C language बहुत से data types provide करती है।

C language के syntax को बहुत सी popular languages ने follow किया है। C का improved version C# है। C के features को Java और C++ जैसी languages ने follow किया है।

Google द्वारा विज्ञापन बंद कर दिया गया
यह विज्ञापन देखना बंद करें
Google द्वारा विज्ञापन ⓘ



Characteristics & Features

- C एक **modular** language है। C में एक बड़े program को छोटे छोटे modules (functions) के रूप में divide किया जाता है। ऐसा करने से programs को manage और debug करने में आसानी रहती है।
- C एक flexible language है। C 32 reserved keywords provide करती है। ये keywords programmer को language पर control provide करते है। इन keywords की मदद से program के structure को भी modify किया जा सकता है।
- C language Assembly language के बाद सबसे fast language मानी जाती है। इसे कई बार low level language भी कहा जाता है। C language hardware के बहुत करीब होती हैं। इसलिए ये दूसरी programming languages से fast होती है।
- C एक case sensitive language है।
- C High level और low level दोनों तरह के programs create करने में सक्षम है।
- C एक extendable language है। C programs में libraries के द्वारा आप कभी भी नए features add कर सकते है।

Limitations

- C language में run time type checking नहीं होती है। C language run time में किसी भी variable के data type को identify करने में असमर्थ होती है। उदाहरण के लिए float data type में integer value डाली जा सकती है।
- C language में object oriented concepts नहीं होते है।
- C language में constructors और destructors का concept भी नहीं होता है।
- C language में namespaces का concept नहीं होता है। इसलिए different programmers के code को एक ही project में यूज़ करना बहुत मुश्किल होता है।
- C language code की re-usability को support नहीं करती है।
- C language में exceptions को run time में handle नहीं किया जा सकता है।

Applications

- ज्यादतर operating systems C language में ही बनाए जाते है। जैसे की UNIX, Windows आदि।
- किसी भी नयी programming language को compile करने के लिए compilers की आवश्यकता होती है। C language में **compilers** भी create किये जाते है। उदाहरण के लिए java का compiler javac भी C language में ही बनाया गया है।
- C language में कई प्रकार के assemblers भी create किये जाते है। Assembler basic instructions को bits में convert करते है।
- C language में text editors भी create किये जा सकते है। जैसे की gedit, आदि।
- C language में network drivers भी create किये जाते है।
- C language में database systems भी लिखे जाते है। जैसे की oracle database C में ही लिखा गया है।

Put Ads on your blog with Adsense

LEARN MORE

C Language

- What is Programming?
- C - Introduction
- C - Tokens, Identifiers & Keywords
- C - Data Types
- C - Variables
- C - First Program
- C - Flow Chart
- C - Operators
- C - Decision Making
- C - Looping
- C - Arrays
- C - Strings
- C - Functions
- C - Structures
- C - Unions
- C - Recursion
- C - Pointers
- C - Type Casting
- C - Storage Classes
- C - Console I/O
- C - File Handling
- C - Error Handling
- C - Header Files
- C - Preprocessor Directives

C Data Types

- Introduction to **C data types in Hindi**
- Different *C data types in Hindi*
  - Integer type in Hindi
  - Floating point type in Hindi
  - Char type in Hindi

Introduction To C Data Types

जब भी आप कोई variable create करते है तो उससे पहले compiler को बताते है की आप किस तरह का data उस variable में store करेंगे। इससे compiler उतनी ही memory उस variable को computer की memory में से allot कर देता है। जैसे की निचे दिए गए statement को देखिये।

```
int age;
```

ऊपर दिया गया statement compiler को बताता है की आप age variable में एक whole number store करने वाले है। किसी भी whole number को store करने के लिए जितनी memory की आवश्यकता होती है compiler उतनी इस variable को allot कर देता है। जो की 2 bytes है।

अब आप सोच सकते है की यदि किसी programming language में data types ना हो तो कितनी memory waste हो सकती है। जँहा पर 2 bytes की requirement हो वँहा 20 bytes आप waste कर सकते है।

इस प्रकार जितनी भी तरह का data आप store कर सकते है उसके लिए पहले से ही maximum memory limit define की गई है। आइये अब C के अलग अलग तरह के data types के बारे में जानने का प्रयास करते है।

Different C data types

C language 3 तरह के data types को support करती है। इन्हें primitive types भी कहते है।

- Integer

Google द्वारा विज्ञापन बंद कर दिया गया
यह विज्ञापन देखना बंद करें
Google द्वारा विज्ञापन ⓘ

    - int
    - short int
    - long int
    - singed int
    - unsigned int
- Floating point
    - float
    - double
- Character
    - char

Integer

Integer types किसी भी whole number (बिना दशमलव के) को store करने के लिए यूज़ किये जाते है। Integer types 5 प्रकार के होते है। हालांकि ये सभी whole number को store करते है। लेकिन memory size और range के base पर इन्हें categories किया गया है।

Data type	Size (Bytes)	Range
int	2	-32768 से 32767
short int	1	-128 से 127
long int	4	-2,147,483,648 से 2,147,483,647
signed int	2	-32768 - 32767
unsigned int	2	0 - 65535

Example

```
long int population = 200000000;
```

Floating point

Floating point data types को दशमलव संख्याओं को store करने के लिए define किया गया है। Floating point data types 2 तरह के होते है। इनको size और range के base पर categories किया गया है। Float type में आप दशमलव के बाद 7 digits तक store कर सकते है। Double type में दशमलव के बाद 17 digits तक store की जा सकती है।

Data types	Size (Bytes)	Range
Float	4	3.4E-38 से 3.4E+38
Double	8	1.7E-308 से 1.7E+308

Example

```
double balance=810.12354984;
```

Character

Character type को एक character store करने के लिए यूज़ किया जाता है। इनको 2 categories में divide किया गया है।

Data type	Size (Bytes)	Range
char	1	-128 से 127
unsigned char	1	0 से 255

Example

```
char bestlanguage = "C";
```

Void type

Void type को उन situations में यूज़ किया जाता है जब आपको value के बारे में कोई जानकारी ना हो। इसे functions के साथ ज्यादातर यूज़ किया जाता है। C में void type को आप इन situations में यूज़ कर सकते है।

1. यदि आपका function कोई value return नहीं करता है तो आप उसका return type void define करते है। उदाहरण के लिए आप इस प्रकार function define कर सकते है। void myFunction();।
2. यदि आप function में कोई parameters नहीं ले रहे है तो आप वँहा पर void define कर सकते है। Void type से पता चलता है की इस function में कोई argument नहीं लिया जाता है। उदाहरण के लिए आप इस प्रकार void को parameter के रूप में pass कर सकते है। int myFunction(void);।
3. यदि आप sure नहीं है की pointer variable किस type के variable को point करेगा तो आप उसका type void declare कर सकते है। इसके बाद आप void pointer से किसी भी variable को point कर सकते है।

© Best Hindi Tutorials 7/18/2016 DMCA PROTECTED

4 Replies so far - Add your comment

Raj Kumar सोमवार, 02 जनवरी, 2017

Intger and character in dono ki limit mujhe samjh me nhi aaya
Like jaise aap ne Floating point me limit 7 and 17 hai wese in dono ke bare me explain kre plz

उत्तर दें

उत्तर

Vipin Sharma सोमवार, 02 जनवरी, 2017

Character data type sirf ek character ko store karta h jese ki "a". Or Integer types ki jitni range upar di gyi h usi range me aap koi bhi sankhya us type ke variable me store kar sakte h.

उत्तर दें

Shubham Gupta शनिवार, 18 फ़रवरी, 2017

Very nice article... thanks for sharing

उत्तर दें

उत्तर

Best Hindi Tutorials शनिवार, 18 फ़रवरी, 2017

You are welcome! Thanks..

उत्तर दें

अपनी टिप्पणी लिखें...

इस रूप में टिप्पणी करें: Google खाता

प्रकाशित करें पूर्वावलोकन

Google Cloud Platform

Build scalable Machine Learning models

TRY IT FREE

C Language

What is Programming?
C - Introduction
C - Tokens, Identifiers & Keywords
C - Data Types
C - Variables
C - First Program
C - Flow Chart
C - Operators
C - Decision Making
C - Looping
C - Arrays
C - Strings
C - Functions
C - Structures
C - Unions
C - Recursion
C - Pointers
C - Type Casting
C - Storage Classes
C - Console I/O
C - File Handling
C - Error Handling
C - Header Files
C - Preprocessor Directives

C Pointers

- Introduction to **C pointers in Hindi**
- Advantages of *pointers in Hindi*
- Disadvantages of pointers in Hindi
- Declaring C pointers in Hindi
- Initializing C pointers in Hindi
- Operators used with C pointers in Hindi
- A complete c pointers Example

Introduction to C pointers

Pointers वो variables होते है जो दूसरे variables के address को store करते है। जैसा की आपको पता है हर variable का memory में एक unique address होता है। ये address hexadecimal form में होता है। इसलिए इस address को आप किसी normal variable में store नहीं कर सकते है। किसी भी variable के address को store करवाने के लिए आप pointer variable create करते है। आइये इसे एक उदाहरण से समझने का प्रयास करते है।

मान लीजिये आपने एक variable create किया है। इस variable का नाम Age है। और आपने इसमें 55 value assign करवायी है। इस variable की memory में location (address) 21F है। अब यदि आप चाहे तो इस address को एक pointer variable में store करवा सकते है। Pointers को यूज़ करने की कुछ advantages निचे दी जा रही है।

Advantages of using pointers

1. Pointers की मदद से आप dynamically memory allocate कर सकते है।
2. Pointers की मदद से आप data structures (linked-list, stack, queue) create कर सकते है।
3. Pointers use करने से program का execution time कम हो जाता है।
4. Pointers की मदद से आप function से एक से अधिक values return कर सकते है।
5. Pointers की मदद से argument passing के दौरान आप variable की copy के बजाय original variable पर काम कर सकते है।
6. Pointers के द्वारा large data को search और sort करना बहुत आसान होता है।

यदि pointers को properly use ना किया जाये तो इसके कुछ disadvantages होते है। इनके बारे में निचे दिया जा रहा है।

Disadvantages of using pointers

1. कई बार pointers की वजह से program में ऐसी error आ जाती है जिसे diagnose करना बहुत difficult होता है।
2. Pointers की वजह से memory में leaks create हो जाते है।
3. यदि run time के दौरान pointers को hold करने के लिए extra memory ना हो तो program crash हो जाता है।
4. Pointers की मदद से restricted memory को access किया जा सकता है।

Working with pointers

यदि आपको एक बार ठीक से समझ आ जाये तो pointers के साथ काम करना बहुत ही आसान है। C language में pointers के साथ काम करने के 2 steps है।

1. Declaring a pointer
2. Initializing a pointer

इन दोनों steps के बारे में निचे detail से समझाया जा रहा है।

Declaring pointers

C में pointers declare करने का general syntax निचे दिया जा रहा है।

```
data_type *pointer_name;
```

C में pointers को declare करने के लिए सबसे पहले आप data type declare करते है। ऐसा इसलिए किया जाता है क्योंकि int type का pointer variable सिर्फ int type के variables का ही address store कर सकता है। इसके बाद आप asterisk (*) operator को declare करते है। इस operator के बाद आप pointer का unique नाम declare करते है। आइये इसे कुछ उदाहरण से समझने का प्रयास करते है।

```
int *int_pntr;                //Pointer for integer variable


double *dbl_pntr;        //Pointer for double variable


char *chr_pntr;            //Pointer for char variable
```

Initializing pointers

C में pointers को initialize करने का syntax निचे दिया जा रहा है।

```
pntr_variable = &variable_name;
```

C में pointers initialize करने के लिए सबसे पहले आप pointer variable को लिखते है। (Initialization part में pointer variable को बिना asterisk operator (*) के लिखा जाता है।) इसके बाद assignment operator (=) लिखा जाता है। इसके बाद आप ampersand operator (&) (इसे address-of operator भी कहा जाता है।) define करते है। इस operator के बाद आप बिना कोई space दिए variable का नाम लिखते है। आइये इसे एक उदाहरण से समझने का प्रयास करते है।

```
int age = 55;                  //Declaring a variable

int *age-pntr;               //Declaring a pointer

age-pntr = &age;        //Initializing a pointer
```

Operators used with pointers

1. & (Address-of) operator - ये operator variable के address को point करता है। इससे से आप किसी भी variable का address प्राप्त कर सकते है।
2. * (Value-at) operator - ये operator सिर्फ pointer variables के साथ काम करता है। ये operator किसी particular address पर store की गयी value को represent करता है।

Example

```
#include<stdio.h>

int main()
{
    int age = 55;
    int *pntr;
    pntr = &age;

    printf("Address of age variable is : %d\n",pntr);

    printf("Value of age variable is : %d",*pntr);

}
```

इस उदाहरण में pointers के द्वारा एक variable का address और उसकी value print करवाई गयी है। ये pointers की working का एक बहुत ही simple उदाहरण है। ये program निचे दिया गया output generate करता है।

```
Address of age variables is : -1074204644;
Value of age variables is : 55
```

© Best Hindi Tutorials 8/13/2016

DMCA PROTECTED

Leave a comment

अपनी टिप्पणी लिखें...

इस रूप में टिप्पणी करें: Google खाता

प्रकाशित करें पूर्वावलोकन

Best Hindi Tutorials

Computer Science and IT Tutorials in Hindi

≡ Menu

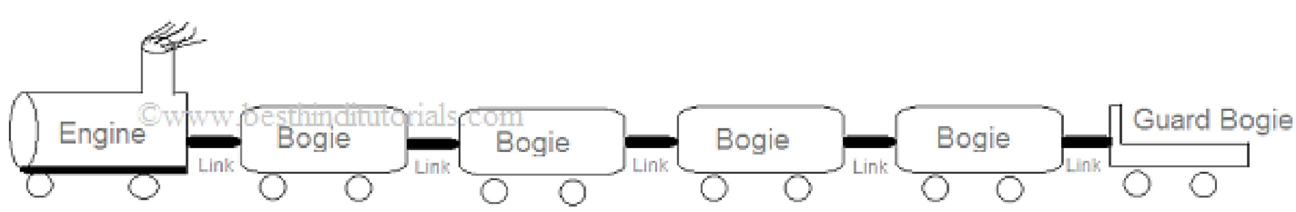
Data Structures in Hindi – Linked List

- Introduction to **linked list in Hindi**
- Array vs *linked list in Hindi*
- Types of linked list in Hindi

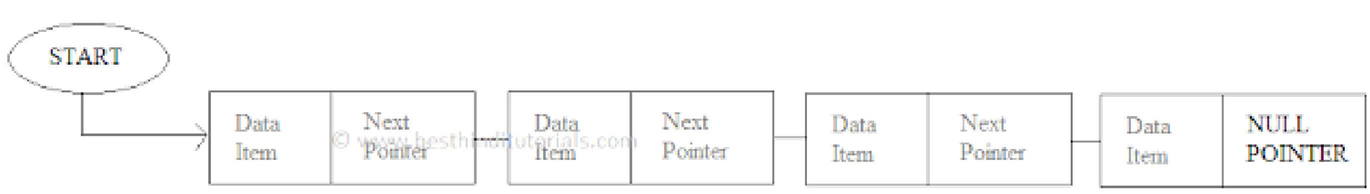
Introduction to Linked List

एक linked list linear (sequential) data structure होती है। जब आप कोई list बनाते है तो उसमे एक के बाद एक item लिखते जाते है। उसी प्रकार linked list में भी एक data item के बाद दूसरा item add किया जाता है।

Linked list को आप किसी train की तरह समझ सकते है। एक train की सभी bogies आपस में connected होती है। आप एक bogie से दूसरी bogie में जा सकते है। उसी प्रकार एक linked list के सभी data items आपस connected होते है।

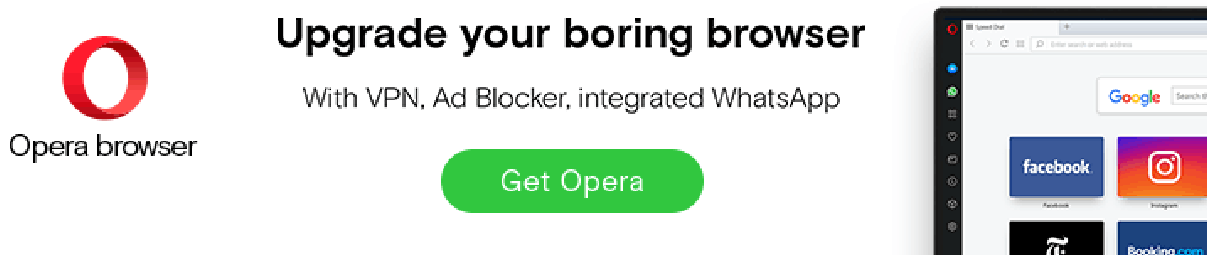


Linked list में add किया गया हर item उससे next item के address को hold करता है और memory में point करता है की next item कँहा पर stored है। इससे अलग अलग locations पर data items stored होने पर भी linked list ऐसे behave करती है जैसे सभी data items एक बाद एक contagious memory locations में store किये गए हो।



जिस प्रकार एक train की शुरआत में engine और अंत में guard bogie होती है। उसी प्रकार एक linked list की शुरआत में start pointer (first node) और अंत में NULL pointer होता है।

C language में programmatically linked list को एक structure द्वारा represent किया जाता है। Structure को आप कोई भी नाम दे सकते है जैसे की node या data आदि। Structure के अंदर दो variables create किये जाते है।



पहला variable data को hold करता है। दूसरा variable define किये जा रहे structure का ही pointer variable होता है। Pointer variable को next या link जैसे नाम दिए जाते है। यह variable next node या data item के address को store करता है।

```
struct Node
{
    int Data;    //Will hold data
    struct Node *Next;  //Will hold address of next node
};
```

Structure का ही pointer variable इसलिए create किया जाता है क्योंकि वह next node का address store करता है और next node भी उसी structure type का variable होती है।

Array vs Linked List

सामन्यतः linked list array की ही तरह प्रतीत होती है। लेकिन कुछ ऐसी असमानताएँ है जो इन दोनों को अलग बनाती है। आइये इन असमानताओं के बारे में जानने का प्रयास करते है।

- एक linked list की size array की तरह fix नहीं होती है। जब आप array create करते है तो साथ ही उसकी size भी define करते है। आप define की गयी size से अधिक elements array में add नहीं कर सकते है। लेकिन linked list की size fix नहीं होती है और आप कितने भी element linked list में add कर सकते है।
- एक linked list में data dynamically store किया जा सकता है। लेकिन array में यह कर पाना असम्भव है।
- एक linked list में आप pointers को re arrange करके data items को आसानी से insert और delete कर सकते है। लेकिन array में insertion और deletion perform करने के लिए data items के बहुत अधिक movement की आवश्यकता होती है।
- एक linked की size को dynamically grow और shrink किया जा सकता है। Array के साथ ऐसा कर पाना संभव नहीं है।
- एक linked list array से अधिक memory का प्रयोग करती है। इसका कारण यह है की linked list में data items के साथ next pointers भी attached होते है।
- एक array के elements को index number द्वारा randomly access किया जा सकता है। लेकिन linked list के elements को आप randomly access नहीं कर सकते है।
- एक linked list के elements को traverse करने में array से अधिक समय लगता है। इसका कारण extra next pointer होता है।
- Linked list data structure में binary searching नहीं perform किया जा सकता है।

- एक array के सभी elements continuous memory locations में store किये जाते है। लेकिन linked list के elements अलग अलग memory locations पर stored हो सकते है। Linked list द्वारा memory का सही utilization किया जाता है।

Types of Linked List

Linked list तीन प्रकार की होती है। इनके बारे में निचे बताया जा रहा है।

Singly Linked List

एक singly linked list सबसे सरल और सामान्य linked list होती है। Singly linked list का हर item उससे अगले item का address store करता है। इस तरह की linked list में सिर्फ एक ही address store किया जाता है। यह भी एक कारण है की इसे singly linked list कहा जाता है।

Doubly Linked list

एक doubly linked list singly linked list की तरह ही होती है। लेकिन doubly linked list में next pointer के अलावा previous pointer भी होता है। Next pointer जँहा अगले element का address store करता है वँही previous pointer previous element का address store करता है।

Doubly linked list में reverse traversal किया जाना संभव होता है।

Circular Linked List

Circular linked list singly linked list ही होती है। लेकिन इस तरह की linked list में आखिर element NULL की बजाय first element का address store करता है। इस प्रकार circular linked list एक loop या cycle की तरह होती है।

Operations with Linked List

एक Linked list के साथ निचे दिए जा रहे operations perform किये जा सकते है।

- Creating a linked list
- Traversing or Displaying a linked list
- Counting nodes of linked list
- Inserting a linked list
- Deleting a linked list
- Searching a linked list

सभी तरह की linked list के साथ ऊपर दिए जा रहे operations perform किये जा सकते है।

Related Posts

Operations on Singly Linked List

ऐसे विभिन्न ऑपरेशन हैं जो singly linked list पर किए जा सकते हैं। ऐसे सभी ऑपरेशनों की एक सूची नीचे दी गई है।

Node Creation :

```
1. struct node
2. {
3.     int data;
4.     struct node *next;
5. };
6. struct node *head, *ptr;
7. ptr = (struct node *)malloc(sizeof(struct node *));
```

Insertion:

एक singly linked list में insertion विभिन्न positions पर की जा सकती है। डाले जा रहे नए node की स्थिति के आधार पर, insertion को निम्नलिखित categories में वर्गीकृत किया गया है।

S.N	Operation	Description
1	Insertion at beginning	इसमें सूची के शुरुआत में किसी भी तत्व को सम्मिलित करना शामिल है। हमें सूची के head के रूप में नया node बनाने के लिए बस कुछ लिंक adjustments की आवश्यकता है।
2	Insertion at the end of the list	इसमें linked list के आखिरी में insertion शामिल है। नए नोड को सूची में एकमात्र नोड के रूप में डाला जा सकता है या इसे last node के रूप में डाला जा सकता है।
3	Insertion after specified node	इसमें linked list के specified नोड के बाद insertion शामिल है। हमें नोड तक पहुंचने के लिए वांछित संख्या के nodes को छोड़ना होगा जिसके बाद नया नोड डाला जाएगा। ।

Deletion and Traversing :

एक singly linked list से नोड का deletion विभिन्न positions पर किया जा सकता है। delete किये जा रहे नोड की position के आधार पर, इस operation को निम्न श्रेणियों में वर्गीकृत किया गया है।

S.N.	Operation	Description
1	Deletion at beginning	इसमें list की शुरुआत से एक नोड को हटाया जाता है। यह इन सब में सबसे सरल operation है। इसके लिए सिर्फ node pointers में कुछ adjustments की जरूरत है।
2	Deletion at the end of the list	इसमें list के अंतिम node को हटाया जाता है। list या तो खाली हो सकती है या भरी हो सकती है। अलग-अलग परिदृश्यों (scenarios) के लिए अलग-अलग तर्क (logic) लागू किए जाते हैं।
3	Deletion after specified node	इसमें list में specified नोड के बाद नोड को हटाया जाता है। हमें नोड तक पहुंचने के लिए वांछित संख्याओं को छोड़ना होगा जिसके बाद नोड हटाया जाता है। इसके लिए list के माध्यम से traversing की आवश्यकता होती है।
4	Traversing	Traversing में, हम सूची के प्रत्येक नोड को कम से कम एक बार उस पर जाते है। ताकि उस पर कुछ विशिष्ट (specific) ऑपरेशन किये जा सके, उदाहरण के लिए, list में मौजूद प्रत्येक नोड के data भाग को print करना।
5	Searching	Searching में, हम दिए गए element के साथ सूची के प्रत्येक element से match करते हैं। यदि element किसी भी location पर पाया जाता है, तो उस element की location आती है अन्यथा null वापस आता है। ।

Google Cloud Platform
Build scalable
Machine Learning models
TRY IT FREE

C Language

What is Programming?
C - Introduction
C - Tokens, Identifiers & Keywords
C - Data Types
C - Variables
C - First Program
C - Flow Chart
C - Operators
C - Decision Making
C - Looping
C - Arrays
C - Strings
C - Functions
C - Structures
C - Unions
C - Recursion
C - Pointers
C - Type Casting
C - Storage Classes
C - Console I/O
C - File Handling
C - Error Handling
C - Header Files
C - Preprocessor Directives

C File Handling

- Introduction to C file handling in Hindi
- Opening a file using **C language in Hindi**
- Closing a file using *C language in Hindi*
- Reading from a file using C language in Hindi
- Writing to a file using C language in Hindi

Introduction to C file handling

एक file bytes की sequence होती है। File किसी भी data को store करने के लिए यूज़ की जाती है। Files को permanent storage के लिए यूज़ किया जाता है। जब आप किसी program को run करवाते है और उसका output देखने के बाद उसे बंद कर देते है तो उसका output वहीं remove हो जाता है। यदि आप इस output को permanently store करना चाहते है तो C language के file handling feature को यूज़ कर सकते है।

C file handling का यूज़ यँही ख़त्म नहीं होता है। यदि program में दिए जाने वाला input भी बहुत अधिक है तो किसी human द्वारा इसे enter कराने के बजाय आप file का यूज़ कर सकते है। इससे program automatically file से ही input ले लेगा।

C language के द्वारा आप files से related ये काम कर सकते है।

1. Existing files को open कर सकते है or नयी files create कर सकते है।
2. किसी भी program के output को file में store कर सकते है।
3. File के द्वारा किसी भी program में input भी दे सकते है।
4. सभी operations perform होने के बाद files को close कर सकते है।

Opening a file

किसी भी file में output store करने या उसमें से input लेने से पहले उसे open किया जाता है। File open करने के लिए आप fopen() function को use करते है।

जब आप किसी ऐसी file को open करने के कोशिश करते है जो exist ही नहीं करती तो वो file automatically create हो जाती है। यदि file पहले से exist करती है तो वह open हो जाती है। आप किस mode में file को open करते है ये उस पर depend करता है।

C language के द्वारा आप files को अलग अलग modes में open कर सकते है। ये mode आप function को call करते समय define करते है। उदाहरण के लिए यदि आप किसी file से सिर्फ data read करने वाले है तो आप उसको read mode में open कर सकते है। और यदि आप file में output store करना चाहते है तो उसे write mode में open कर सकते है। और आप चाहे तो किसी file को दोनों modes में भी open कर सकते है।

File opening modes

File open करते समय यूज़ किये जाने वाले सभी modes के बारे में नीचे दिया जा रहा है।

Mode	Description
r	इस mode से आप एक existing file को open कर सकते है। इस mode में आप files को सिर्फ read कर सकते है।
w	इस mode से आप सिर्फ files में write कर सकते है। यदि file पहले से exist नहीं करती है तो वह create हो जाती है।
a	इस mode से आप output write करते है। इस mode में open की गयी file यदि exist करती है तो वह open हो जाती है और उसमें जो data होता है उसी के बाद से नया data add हो जाता है। यदि file exist नहीं करती है तो नयी file create हो जाती है।
r+	इस mode में आप एक file को reading और writing दोनों purposes के लिए open कर सकते है।
w+	इस mode में file reading और writing दोनों purposes के लिए open की जाती है। Write करने के लिए इस mode में पहले file zero length तक truncate की जाती है।
a+	इस mode में file reading और writing दोनों modes में open होती है। यदि file exist नहीं करती तो create हो जाती है। और writing हमेशा existing data के बाद शुरू होती है।

fopen() function का general structure निचे दिया जा रहा है।

```
FILE *fopen(const char * file_name, const char * mode );
```

इस function में 2 arguments pass किये जाते है। एक तो hard disk पर file का address और दूसरा वह mode जिसमें आप file open करना चाहते है। ये function FILE type का एक object return करता है। इसलिए इसका return type FILE होता है। और इसी लिए आप इस function के द्वारा return किये गए object को FILE type के दूसरे object में store करेंगे। इसका उदाहरण आप आगे देखेंगे।

Reading from a file

C language के द्वारा किसी file को read करने के लिए आप 2 functions यूज़ कर सकते है।

1. fgetc()
2. fgets()

fgetc() function एक बार में एक ही character read करता है लेकिन इसे आप loop में यूज़ करके पूरी file को read कर सकते है। इस function में open की गयी file के pointer के रूप में एक ही argument pass किया जाता है। इसका उदाहरण नीचे दिया जा रहा है।

```
#include<stdio.h>

int main()
{
    FILE *fp;
    char data[1];
    fp = fopen("input.txt","r");
    data=fgetc(fp);
    printf("%s",data[]);
    fclose(fp);

    return 0;
}
```

उपर दिए गए उदाहरण में fgetc() function का यूज़ करते हुए file में से एक character read किया गया है। ये program input.txt file से एक character read करता है। यदि file exist नहीं करती है तो run time error generate होती है।

fgets function से आप define किये गए characters तक file को read कर सकते है। इस function में 3 arguments pass किये जाते है। पहला argument एक character array होता है। ये वो array होता है जिसमे read किया गया data store किया जाता है। दूसरा argument जितने characters आप read करना चाहते है उनकी size होती है। तीसरा argument file pointer होता है। इसका उदाहरण नीचे दिया जा रहा है।

```
#include<stdio.h>

int main()
{
    FILE *fp;
    char data[100];
    fp=fopen("test.txt","r");
    fgets(data,100,fp);
    printf("%s",data);
    fclose(fp);

    return 0;
}
```

ऊपर दिया गया program test.txt file से 100 characters read करता है। यदि file पहले से exist नहीं करती है तो run time error generate होती है।

Writing to a file

C language के द्वारा किसी file में data write करने के लिए आप 2 functions यूज़ कर सकते है।

1. fputc()
2. fputs()

यँहा पर fputs() function का example नीचे दिया जा रहा है। इस function में 2 arguments pass किये जाते है। पहला argument वो string होती है जिसे आप file में store करना चाहते है। और दूसरा argument file pointer होता है।

```
#include <stdio.h>

int main()
{
    FILE *fp;
    fp = fopen("test.txt","w+");
    fputs("This is text string",fp);
    fclose(fp);

    return 0;
}
```

ऊपर दिया गया program test.txt file में This is a string write करता है।

Closing a file

File को close करने के लिए आप fclose() function को यूज़ करते है। इसका general structure नीचे दिया जा रहा है।

```
int fclose(FILE *fp);
```

उपर दिए गए function में *fp वो FILE type object है जिसमे आपने fopen() function के द्वारा return किये गए object को store किया था।

© Best Hindi Tutorials 8/27/2016 DMCA PROTECTED

Leave a comment

अपनी टिप्पणी लिखें...

इस रूप में टिप्पणी करें: Google खाता ▼

प्रकाशित करें पूर्वावलोकन