
অপারেটর

লেকচার-১১

অপারেটর

লেকচার-১১

এই পাঠ শেষে যা যা শিখতে পারবে-

- ১। অপারেটর ব্যাখ্যা করতে পারবে।
- ২। 'সি' প্রোগ্রামিং ভাষার বিভিন্ন অপারেটর এবং তাদের ব্যবহার বর্ণনা করতে পারবে।
- ৩। রাশিমালা ব্যাখ্যা করতে পারবে।
- ৪। অপারেটরের precedence এবং associativity ব্যাখ্যা করতে পারবে।

অপারেটর

'সি' প্রোগ্রামিং ভাষায় গাণিতিক এবং যৌক্তিক কাজ সম্পাদন করার জন্য কতগুলো বিশেষ চিহ্ন বা সিম্বল ব্যবহৃত হয়, এই সিম্বল বা চিহ্নগুলোকে অপারেটর বলা হয়। অপারেটরগুলো যার উপর কাজ করে তাকে অপারেণ্ড বলা হয়। যেমনঃ $A + B * 5$ এই এক্সপ্রেশনটিতে $+$, $*$ হলো অপারেটর ও A , B হলো অপারেণ্ড, 5 হলো ধ্রুবক এবং $A + B * 5$ হলো এক্সপ্রেশন।

অপারেটর কতগুলো অপারেণ্ড নিয়ে কাজ করে তার উপর ভিত্তি করে তিন প্রকার। যথা-

- ১। ইউনারি(Unary) অপারেটর
- ২। বাইনারি(Binary) অপারেটর
- ৩। টারনারি(Ternary) অপারেটর

ইউনারি(Unary) অপারেটরঃ যেসব অপারেটর শুধুমাত্র একটি অপারেণ্ড নিয়ে কাজ করে তাদেরকে

ইউনারি(Unary) অপারেটর বলে। যেমনঃ Increment ($++$) and decrement ($-$) operators

বাইনারি(Binary) অপারেটরঃ যেসব অপারেটর দুইটি অপারেণ্ড নিয়ে কাজ করে তাদেরকে বাইনারি(Binary) অপারেটর বলে। যেমনঃ

1. Arithmetic operators ($+$, $-$, $*$ etc.)
2. Relational Operators ($<$, $>$, $==$)
3. Logical Operators ($\&\&$, $\|\|$)
4. Assignment Operators ($=$, $+=$, $-$ $=$)
5. Bitwise Operators ($\&$, $\|$)

টারনারি(Ternary) অপারেটরঃ যেসব অপারেটর তিনটি অপারেণ্ড নিয়ে কাজ করে তাদেরকে টারনারি(Ternary) অপারেটর বলে। যেমনঃ Conditional Operators($?:$)

কাজের প্রকৃতির উপর ভিত্তি করে 'সি' প্রোগ্রামিং ভাষার অপারেটর সমূহ:

- ১। গাণিতিক অপারেটর (Arithmetic Operators)
- ২। রিলেশনাল অপারেটর (Relational Operators)
- ৩। লজিক্যাল অপারেটর (Logical Operators)

৪। অ্যাসাইনমেন্ট অপারেটর (Assignment Operators)

৫। ইনক্রিমেন্ট এবং ডিক্রিমেন্ট অপারেটর (Increment and Decrement Operators)

৬। কন্ডিশনাল অপারেটর (Conditional Operators)

৭। বিট ওয়াইজ অপারেটর (Bitwise Operators)

৮। বিশেষ অপারেটর (Special Operator)

হয়। রিলেশন বলতে একটি অপারেটর অপার অপারেটর থেকে ছোট কিংবা বড় বা সমান ইত্যাদি বুঝায়।

রিলেশনাল অপারেটর	কাজ	উদাহরণ	বর্ণনা
<	Less than (ছোট)	$a < b$	a এর মান b মানের চেয়ে ছোট
<=	Less than or equal (ছোট বা সমান)	$a <= b$	a এর মান b মানের চেয়ে ছোট অথবা সমান
>	Greater than (বড়)	$a > b$	a এর মান b মানের চেয়ে বড়
>=	Greater than or equal (বড় বা সমান)	$a >= b$	a এর মান b মানের চেয়ে বড় অথবা সমান।
==	Equal to (সমান)	$a == b$	a এর মান b মানের সমান
!=	Not equal to (অসমান)	$a != b$	a ও b এর মান সমান নয়

লজিক্যাল অপারেটর (Logical Operators): প্রোগ্রামে যুক্তিমূলক এক্সপ্রেশন নিয়ে কাজ করার জন্য যেসব অপারেটর ব্যবহার করা হয় সেগুলোকে লজিক্যাল অপারেটর বলা হয়।

অপারেটর এর চিহ্ন	অপারেটর এর নাম	কার্যপদ্ধতি	উদাহরণ
&&	AND অপারেটর	যদি উভয় অপারেটর এর মান শূন্য না হয় তবেই শর্তটি সত্য বা true হবে।	(A && B) is true.
	OR অপারেটর	যদি দুটি অপারেটর এর কমপক্ষে একটি মান শূন্য না হয় তবেই শর্তটি সত্য বা true হবে।	(A B) is true.
!	NOT অপারেটর	অপারেটর এর মান বিপরীত অর্থে ব্যবহৃত হয়। যদি একটি শর্ত সত্য বা true হয় এবং সেক্ষেত্রে লজিক্যাল NOT অপারেটর ব্যবহারের ফলে শর্তটি মিথ্যা বা false হবে।	!(A && B) is false.

গাণিতিক অপারেটর (Arithmetic Operators): 'সি' প্রোগ্রামে বিভিন্ন গাণিতিক কাজ (যেমন-যোগ, বিয়োগ, গুণ, ভাগ প্রভৃতি) করার জন্য যেসব অপারেটর ব্যবহৃত হয়, সেসব অপারেটরকে গাণিতিক অপারেটর বলা হয়।

অপারেটর (Operator)	নাম (Name)	ব্যবহার (Uses)
+	plus	যোগ করার জন্য ব্যবহৃত হয়।
-	minus	বিয়োগ করার জন্য ব্যবহৃত হয়।
/	division	ভাগ করে ভাগফল নির্ণয়ের জন্য ব্যবহৃত হয়।
*	multiplier	গুণ করার জন্য ব্যবহৃত হয়।
%	modulus	ভাগশেষ বের করার জন্য ব্যবহৃত হয়।

'সি' প্রোগ্রামিং ভাষায় গাণিতিক অপারেটর গুলোর অগ্রগণ্যতা হলঃ *, /, %, +, -

অ্যাসাইনমেন্ট অপারেটর (Assignment Operators): কোনো এক্সপ্রেশন বা ভেরিয়েবলের মানকে অন্য কোনো ভেরিয়েবলের মান হিসেবে নির্ধারণ করতে যেসব অপারেটর ব্যবহার করা হয়, সেগুলোকে অ্যাসাইনমেন্ট অপারেটর বলা হয়।

রিলেশনাল অপারেটর (Relational Operators): প্রোগ্রাম নির্বাহের সময় দুটি চলকের মধ্যে তুলনার ক্ষেত্রে রিলেশনাল অপারেটর ব্যবহৃত

Simple assignment operator	Short hand assignment operator
a = a +	a += 1
a = a - 1	a -= 1
a = a * b	a *= b
a = a/b	a /= b
a = a%b	a %= b

ইনক্রিমেন্ট এবং ডিক্রিমেন্ট অপারেটর (Increment and Decrement Operators):

‘সি’ প্রোগ্রামে দুটি গুরুত্বপূর্ণ অপারেটর ব্যবহার করা হয় যা সাধারণত অন্য ভাষায় ব্যবহার করা হয় না। অপারেটর দুটি হচ্ছে Increment (++) and Decrement (--) Operators। ইনক্রিমেন্ট অপারেটর ব্যবহার করা হয় কোন একটি ভেরিয়েবলের মান ১ বৃদ্ধি করতে এবং ডিক্রিমেন্ট অপারেটর ব্যবহার করা হয় কোন একটি ভেরিয়েবলের মান ১ হ্রাস করতে। ইনক্রিমেন্ট এবং ডিক্রিমেন্ট উভয় অপারেটর একটি অপারেন্ডের উপর কাজ করে। তাই এদেরকে ইউনারি অপারেটর বলা হয়।

ইনক্রিমেন্ট অপারেটরের প্রকারভেদঃ

- pre-increment
- post-increment

pre-increment(++ variable): pre ইনক্রিমেন্ট এর ক্ষেত্রে চলকের মান আগে বৃদ্ধি করে এবং তারপর আপডেট মানটি নিয়ে কাজ করে।

```
#include<stdio.h>
#include<conio.h>

main()
{
    int x,i;
    i=10;
    x=++i;
    printf("x: %d",x);
    printf("i: %d",i);
    getch();
}
```

Output:

x: 11

i: 11

post-increment (variable ++): post ইনক্রিমেন্ট এর ক্ষেত্রে চলকের বর্তমান মান নিয়ে কাজ করে তারপর চলকের মান বৃদ্ধি করে।

```
#include<stdio.h>
#include<conio.h>

void main()
{
    int x,i;
    i=10;
    x=i++;
    printf("x: %d",x);
    printf("i: %d",i);
    getch();
}
```

Output:

x: 10

i: 11

```
#include<stdio.h>
#include<conio.h>

void main()
{
    int x,i;
    i=10;
    x=--i;
    printf("x: %d",x);
    printf("i: %d",i);
    getch();
}
```

Output:

x: 9

i: 9

ডিক্রিমেন্ট অপারেটরের প্রকারভেদঃ

- pre-decrement
- post-decrement

post-decrement (variable -): post ডিক্রিমেন্ট এর ক্ষেত্রে চলকের বর্তমান মান নিয়ে কাজ করে তারপর চলকের মান হ্রাস করে।

Pre-decrement (- variable): pre ডিক্রিমেন্ট এর ক্ষেত্রে চলকের মান আগে হ্রাস করে এবং তারপর আপডেট মানটি নিয়ে কাজ করে।

```
#include<stdio.h>
#include<conio.h>
```

```
void main()
{
    int x,i;
    i=10;
    x=i--;
    printf("x: %d",x);
    printf("i: %d",i);
    getch();
}
```

Output:

x: 10

i: 9

উপরের উদাহরণে, যদি A, 0 এর চেয়ে ছোট হয় তাহলে Negative রিটার্ন করবে অন্যথায় Positive রিটার্ন করবে।



কন্ডিশন সত্য হলে Expr1 সম্পাদিত হবে এবং কন্ডিশন মিথ্যা হলে Expr2 সম্পাদিত হবে। এই অপারেটরকে টারনারি অপারেটরও বলা হয়।

বিট ওয়াইজ অপারেটর (Bitwise Operators):

‘সি’ প্রোগ্রামে বিট পরীক্ষা করার জন্য বা কোনো বিট ডানে বা বামে সরানোর জন্য বিট ওয়াইজ অপারেটর ব্যবহার করা হয়। Float বা Double টাইপের ডেটার ক্ষেত্রে বিট ওয়াইজ অপারেটর ব্যবহার করা যায় না।

Operator	Meaning	Example
&	Bitwise AND	a&b
	Bitwise OR	a b
^	Bitwise Exclusive OR	a^b
~	Bitwise NOT	~a
<<	Left Shift	a<<1
>>	Right Shift	a>>1

কন্ডিশনাল অপারেটর (Conditional Operators):

‘সি’ প্রোগ্রামে শর্ত সাপেক্ষে কোন কাজ করার জন্য কন্ডিশনাল অপারেটর ব্যবহৃত হয়। কন্ডিশনাল অপারেটরের গঠন নিম্নরূপঃ

Syntax : (Condition? true_value: false_value);

Example : (A < 0 ? Negative : Positive);

বিট ওয়াইজ অপারেটর এর সত্যক সারণিঃ

A	B	A B	A & B	A ^ B	~A
0	0	0	0	0	1
0	1	1	0	1	1
1	0	1	0	1	0
1	1	1	1	0	0

বিশেষ অপারেটর (Special Operator): 'সি' প্রোগ্রামে বিশেষ কিছু কাজের জন্য ব্যবহৃত অপারেটরকে বিশেষ অপারেটর বলে। যেমনঃ

অপারেটর	বর্ণনা
&	চলকের অ্যাড্রেস পেতে এই অপারেটর ব্যবহৃত হয়। উদাহরণ : &a, a এর অ্যাড্রেস দিবে।
*	চলকের পয়েন্টার হিসেবে ব্যবহৃত হয়। উদাহরণ : * a যেখানে, * হলো a চলকের পয়েন্টার।
Sizeof ()	চলকের মেমোরি সাইজ রিটার্ন করে। উদাহরণ: size of (char); ১ রিটার্ন করবে।

রাশিমালা (Expression)

চলক, ধ্রুবক ও বিভিন্ন অপারেটরের সমন্বয়ে রাশিমালা বা Expression তৈরি হয়। গাণিতিক Expression গুলো 'সি' প্রোগ্রামে নিম্নরূপে লেখা হয়-

গাণিতিক এক্সপ্রেশন	'সি' ভাষায় এক্সপ্রেশন
ax^2+bx+c	$a*x*x+b*x+c$
$Y=a^3+b^3+c^3$	$Y=a*a*a+b*b*b+c*c*c$
$F=x+\frac{y}{z}+c$	$F=x+(y/z)+c$
$Y=(a^n)^m$	$Y=pow(pow(a,n),m)$
$Y=\sqrt{b^2-4ac}$	$Y=sqrt(b*b-4*a*c)$
$Y= a-b +c$	$Y=abs(a-b)+c$

Expression বা রাশিমালায় অপারেটরের precedence এবং associativity:

একটি রাশিমালায় একাধিক অপারেটর থাকলে কম্পাইলার অপারেটরগুলোর অগ্রগণ্যতা দেয় এবং অগ্রগণ্যতা অনুযায়ী কাজ করে। প্রোগ্রামিং ভাষায় একে অপারেটরের precedence বলে। 'সি' প্রোগ্রামিং ভাষায় গাণিতিক অপারেটর গুলোর অগ্রগণ্যতা হলঃ *, /, %, +, - । অর্থাৎ $x+y*x-a$ এই রাশিমালায় আগে গুণের কাজ, তারপর যোগের কাজ এবং সবশেষে বিয়োগের কাজ সম্পন্ন হবে।

আবার একটি রাশিমালায় একই precedence বিশিষ্ট একাধিক অপারেটর থাকলে কম্পাইলার কিছু অপারেটরের ক্ষেত্রে বামদিক থেকে ডানদিকে এবং কিছু অপারেটরের ক্ষেত্রে ডানদিক থেকে বামদিকে কাজ করে। প্রোগ্রামিং ভাষায় একে অপারেটর associativity বলে। যেমন- $x+y+z+a$ রাশিমালায় '+' অপারেটর একাধিকবার ব্যবহৃত হয়েছে। এই '+' অপারেটরের associativity হল বাম থেকে ডান। তাই উপরের রাশিমালায় প্রথমে x ও

y যোগ করে তার সাথে z এর যোগ এবং অবশেষে প্রাপ্ত যোগফলের সাথে a এর যোগ।

নিচের টেবিলে, সর্বোচ্চ precedence (অগ্রাধিকার) বিশিষ্ট অপারেটর টেবিলের শীর্ষে অবস্থিত এবং সর্বনিম্ন precedence (অগ্রাধিকার) বিশিষ্ট অপারেটর টেবিলের সর্বনিম্ন নীচে অবস্থিত।

Category	Operator	Associativity
Postfix	() [] -> . ++ --	Left to right
Unary	+ - ! ~ ++ -- (type)* & sizeof	Right to left
Multiplicative	* / %	Left to right
Additive	+ -	Left to right
Shift	<< >>	Left to right
Relational	< <= > >=	Left to right
Equality	== !=	Left to right
Bitwise AND	&	Left to right
Bitwise XOR	^	Left to right
Bitwise OR		Left to right
Logical AND	&&	Left to right
Logical OR		Left to right
Conditional	?:	Right to left
Assignment	= += -= *= /= %=>>= <<= &= ^= =	Right to left
Comma	,	Left to right

পাঠ মূল্যায়ন-

জ্ঞানমূলক প্রশ্নসমূহ

১। অপারেটর কী?

উত্তরঃ প্রোগ্রামিং ভাষায় গাণিতিক এবং যৌক্তিক কাজ নিয়ন্ত্রণ করার জন্য কতগুলো বিশেষ চিহ্ন ব্যবহৃত হয়, এগুলোকে অপারেটর বলা হয়। যেমন- $+$, $-$, $*$, $\%$ ইত্যাদি।

২। ইউনারি অপারেটর কী?

উত্তরঃ যেসব অপারেটর শুধুমাত্র একটি অপারেণ্ড নিয়ে কাজ করে তাদেরকে ইউনারি(Unary) অপারেটর বলে।

৩। বাইনারি অপারেটর কী?

উত্তরঃ যেসব অপারেটর দুইটি অপারেণ্ড নিয়ে কাজ করে তাদেরকে বাইনারি(Binary) অপারেটর বলে।

৪। টারনারি অপারেটর কী?

উত্তরঃ যেসব অপারেটর তিনটি অপারেণ্ড নিয়ে কাজ করে তাদেরকে টারনারি(Ternary) অপারেটর বলে।

৫। লজিক্যাল অপারেটর কী?

উত্তরঃ প্রোগ্রামে যুক্তিমূলক এক্সপ্রেশন নিয়ে কাজ করার জন্য যেসব অপারেটর ব্যবহার করা হয় সেগুলোকে লজিক্যাল অপারেটর বলা হয়।

৬। অপারেণ্ড কী?

উত্তরঃ অপারেটরগুলো যার উপর কাজ করে তাকে অপারেণ্ড বলা হয়।

৭। অপারেটর প্রিসিডেন্স কী?

উত্তরঃ একটি রাশিমালায় একাধিক অপারেটর থাকলে কম্পাইলার অপারেটরগুলোর অগ্রগণ্যতা দেয় এবং অগ্রগণ্যতা অনুযায়ী কাজ করে। প্রোগ্রামিং ভাষায় একে অপারেটরের precedence বলে।

অনুধাবনমূলক প্রশ্নসমূহ

১। কখন ইউনারি অপারেটর ব্যবহার করা হয়? ব্যাখ্যা কর।

২। প্রোগ্রামে অপারেটরের গুরুত্ব লেখ।

৩। $i++$ এবং $++i$ ব্যাখ্যা কর।

৪। $=$ এবং $==$ এর মধ্যে পার্থক্য লেখ।