
কন্ট্রোল স্টেটমেন্ট

লেকচার-১৪

কন্ট্রোল স্টেটমেন্ট

লেকচার-১৪

এই পাঠ শেষে যা যা শিখতে পারবে-

- ১। কন্ট্রোল স্টেটমেন্ট ব্যাখ্যা করতে পারবে।
- ২। বিভিন্ন প্রকার কন্ট্রোল স্টেটমেন্ট ব্যাখ্যা করতে পারবে।
- ৩। কন্ডিশনাল কন্ট্রোল স্টেটমেন্ট সম্পর্কে বিস্তারিত ব্যাখ্যা করতে পারবে।
- ৪। কন্ডিশনাল কন্ট্রোল স্টেটমেন্ট ব্যবহার করে প্রোগ্রাম লিখতে পারবে।

কন্ট্রোল স্টেটমেন্ট

‘সি’ প্রোগ্রামিং ভাষায় স্টেটমেন্টসমূহ সাধারণত স্বয়ংক্রিয়ভাবে পর্যায়ক্রমে নির্বাহ হয়। কিন্তু বিভিন্ন পরিস্থিতিতে প্রোগ্রামের নির্বাহ নিয়ন্ত্রণ (যেমন- এক বা একাধিক স্টেটমেন্ট একাধিক বার নির্বাহ, শর্ত সাপেক্ষে কোন এক বা একাধিক স্টেটমেন্ট নির্বাহ, এক স্টেটমেন্ট থেকে অন্য স্টেটমেন্টে প্রোগ্রামের নিয়ন্ত্রণ স্থানান্তর ইত্যাদি) করার প্রয়োজন হয়। যে সকল স্টেটমেন্ট এর সাহায্যে প্রোগ্রাম স্টেটমেন্টসমূহের পর্যায়ক্রমিক নির্বাহ নিয়ন্ত্রণ করা যায়, তাদেরকে কন্ট্রোল স্টেটমেন্ট বলে।

কন্ট্রোল স্টেটমেন্ট সমূহঃ

- ১। কন্ডিশনাল কন্ট্রোল স্টেটমেন্ট/ ডিসিশন কন্ট্রোল স্টেটমেন্ট
- ২। লুপ কন্ট্রোল স্টেটমেন্ট
- ৩। জাম্পিং কন্ট্রোল স্টেটমেন্ট

কন্ডিশনাল কন্ট্রোল স্টেটমেন্ট

‘সি’ প্রোগ্রামে শর্তসাপেক্ষে এক বা একাধিক স্টেটমেন্ট নির্বাহের জন্য কন্ডিশনাল কন্ট্রোল স্টেটমেন্ট ব্যবহৃত হয়। কন্ডিশনাল কন্ট্রোল স্টেটমেন্টে ব্যবহৃত শর্ত সত্য হলে প্রোগ্রামে এক ধরনের ফলাফল পাওয়া যায় এবং মিথ্যা হলে অন্য ধরনের ফলাফল পাওয়া যায়।

‘সি’ প্রোগ্রামিং ভাষায় কন্ডিশনাল কন্ট্রোল স্টেটমেন্টসমূহঃ

- ১। if স্টেটমেন্ট
- ২। if-else স্টেটমেন্ট
- ৩। else if স্টেটমেন্ট
- ৪। nested if-else স্টেটমেন্ট
- ৫। switch স্টেটমেন্ট

if স্টেটমেন্টঃ প্রোগ্রামে শর্ত সাপেক্ষে এক বা একাধিক স্টেটমেন্ট নির্বাহের জন্য if স্টেটমেন্ট ব্যবহার করা হয়। এক্ষেত্রে if স্টেটমেন্ট তার কন্ডিশনটি চেক করে। যদি কন্ডিশন সত্য হয় তাহলে বডি়র মধ্যে অবস্থিত স্টেটমেন্টসমূহ নির্বাহ হয়। আর যদি কন্ডিশন মিথ্যা হয় তাহলে বডি়র মধ্যে অবস্থিত স্টেটমেন্টসমূহ নির্বাহ হয় না। if স্টেটমেন্ট ব্যবহারের ফরম্যাট হলো-

```
if(condition)
{
    statement(s)
}
```

কীবোর্ড থেকে কোনো সংখ্যা ইনপুট দিয়ে দেখবে সংখ্যাটি ধনাত্মক কিনা?

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

main()
{
    int a;
    printf("Enter a number");
    scanf("%d",&a);
    if(a>0)
        printf("The given number is Positive");
    getch();
}
```

কীবোর্ড থেকে কোনো সংখ্যা ইনপুট দিয়ে দেখবে সংখ্যাটি ধনাত্মক না ঋনাত্মক।

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

main()
{
    int a;
    printf("Enter a number");
    scanf("%d",&a);
    if(a>=0)
        printf("The given number is Positive");
    else
        printf("The given number is Negative");
    getch();
}
```

if-else স্টেটমেন্ট: if-else স্টেটমেন্টের ক্ষেত্রে if এর কন্ডিশনটি সত্য হলে নির্দিষ্ট স্টেটমেন্টসমূহ নির্বাহ হয়। আর যদি প্রোগ্রামের কোন কন্ডিশন সত্য না হয়, তাহলে বস্তুৎ এর স্টেটমেন্টসমূহ নির্বাহ হয়। ‘সি’ প্রোগ্রামে ‘অন্যথায়’ অর্থে else স্টেটমেন্ট ব্যবহৃত হয়। else স্টেটমেন্টে কোন কন্ডিশন থাকে না। if-else স্টেটমেন্ট ব্যবহারের ফরম্যাট হলো-

```
if(condition)
{
    statement(s)
}
else
{
    statement(s)
}
```

else if স্টেটমেন্ট: প্রোগ্রামে যদি একাধিক কন্ডিশন যাচাই করতে হয় তাহলে প্রথম কন্ডিশন যাচাই করার জন্য if স্টেটমেন্ট ব্যবহার করা হয়। তারপরের কন্ডিশন গুলো যাচাই করার জন্য else if স্টেটমেন্ট ব্যবহার করা হয়। ‘সি’ প্রোগ্রামে “অন্যথায় যদি” অর্থে বস্তুৎ রত স্টেটমেন্ট ব্যবহার করা হয়। else if স্টেটমেন্ট ব্যবহারের ফরম্যাট হলো-

```

if(Condition1)
{
    statement(s)
}
else if(Condition2)
{
    statement(s)
}
.....
.....

else
{
    statement(s)
}

```

nested if-else স্টেটমেন্টঃ একটি if-else স্টেটমেন্টের মধ্যে যখন অন্য এক বা একাধিক if-else স্টেটমেন্ট ব্যবহৃত হয় তখন তাকে nested if-else স্টেটমেন্ট বলে। nested if-else স্টেটমেন্ট ব্যবহারের ফরম্যাট হলো-

```

if(Condition1)
{
    if(Condition2)
    {
        statement(s)
    }
}

```

লিপ ইয়ার(অধিবর্ষ) নির্ণয়ের প্রোগ্রাম।

কীবোর্ড থেকে কোনো সংখ্যা ইনপুট দিয়ে দেখবে সংখ্যাটি শূন্য, ধনাত্মক অথবা ঋনাত্মক।

```

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

main()
{
    int a;
    printf("Enter a number");
    scanf("%d",&a);
    if(a==0)
        printf("The given number is Zero");
    else if(a>0)
        printf("The given number is Positive");
    else
        printf("The given number is Negative");
    getch();
}

```

```

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

main()
{
    int y;
    printf("Enter a year");
    scanf("%d",&y);
    if(y%4==0)
    {
        if(y%100!=0)
            printf("The given year is a leap year");
        }
    else if(y%400==0)
        printf("The given year is a leap year");
    else
        printf("The given year is not a leap year");
    getch();
}

```

নোটঃ প্রতিটি কন্ট্রোল স্টেটমেন্টের জন্য একাধিক স্টেটমেন্ট ব্যবহৃত হলে স্টেটমেন্টসমূহ কার্লিব্রেস { } এর মধ্যে লিখতে হয়। আর যদি একটি স্টেটমেন্ট ব্যবহৃত হয় তাহলে কার্লিব্রেস { } এর মধ্যে না লিখলেও সমস্যা নেই। উপরের উদাহরণে প্রতিটি কন্ট্রোল স্টেটমেন্টের জন্য একটি স্টেটমেন্ট লেখা আছে তাই কার্লিব্রেস { } এর মধ্যে লেখা হয় নি।

switch স্টেটমেন্ট

প্রোগ্রামে একাধিক কন্ডিশন লেখার ক্ষেত্রে else if স্টেটমেন্ট ব্যবহার করা হয়। অনেক বড় প্রোগ্রামের ক্ষেত্রে else if স্টেটমেন্ট ব্যবহার করে প্রোগ্রাম লেখা কষ্টকর। তাই এই ক্ষেত্রে else if স্টেটমেন্ট এর পরিবর্তে switch স্টেটমেন্ট ব্যবহার করে প্রোগ্রাম লেখা সহজ। switch স্টেটমেন্ট ব্যবহারের ফরম্যাট হলো-

```
switch(expression)
{
    case constant_1:
        statements_for_constant_1;
        break;

    case constant_2:
        statements_for_constant_2;
        break;

    -----

    case constant_n:
        statements_for_constant_n;
        break;

    default: // Default case is optional
        default_statements;
        break;
}
```

switch স্টেটমেন্টের expression এর মানের সাথে যে case constant টি ম্যাচিং করবে সেই case এর কোড ব্লকটি নির্বাহ হবে। যদি কোন case constant এর সাথে না মিলে তাহলে default কোড ব্লকটি নির্বাহ হবে।

switch স্টেটমেন্ট সম্পর্কে গুরুত্বপূর্ণ তথ্যঃ

১। switch স্টেটমেন্টে case constant গুলোর মান char('A', 'B'.../ 'a', 'b'.../'+', '- '...ইত্যাদি) অথবা int(1,2,3...) টাইপ হতে হবে। case constant এর মান char টাইপের হলে তা সিঙ্গেল কোটেশনের(' ') মধ্যে লিখতে হয়।

২। একটি switch স্টেটমেন্টের case constant গুলোর মান অবশ্যই ভিন্ন ভিন্ন হতে হবে।

৩। case constant গুলোর শেষে কোলন (:) থাকতে হবে।

৪। প্রতিটি case এর শেষে break কিওয়ার্ড থাকতে হবে।

৫। একাধিক case এর জন্য একই কোড ব্লক হতে পারবে।

৬। একটি মাত্র ডিফল্ট case থাকতে পারে।

উদাহরণ-২ঃ একটি অক্ষর স্বরবর্ণ না ব্যঞ্জনবর্ণ তা নির্ণয়ের জন্য switch স্টেটমেন্টের সাহায্যে একটি প্রোগ্রাম লিখ।

উদাহরণ-১ঃ দুইটি সংখ্যা ইনপুট নিয়ে switch স্টেটমেন্টের সাহায্যে যোগ অথবা বিয়োগ অথবা গুন অথবা ভাগ করার একটি প্রোগ্রাম।

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

main()
{
    char ope;
    int a,b;
    printf("Enter Operator Either + or - or * or /");
    scanf("%c",&ope);
    printf("Enter Two numbers");
    scanf("%d %d",&a,&b);

    switch(ope)
    {
        case '+':
            printf("Sum=%d",a+b);
            break;
        case '-':
            printf("Subtraction=%d",a-b);
            break;
        case '*':
            printf("Multiplication=%d",a*b);
            break;
        case '/':
            printf("Quotient=%d",a/b);
            break;
        default:
            printf("Your Operation is not matched");
            break;
    }
    getch();
}
```

পাঠ মূল্যায়ন-

জ্ঞানমূলক প্রশ্নসমূহ

১। কন্ট্রোল স্টেটমেন্ট কী?

উত্তরঃ সি প্রোগ্রামের স্টেটমেন্টসমূহ সাধারণত পর্যায়ক্রমিকভাবে নির্বাহ হয়। কিন্তু কখনও কখনও স্টেটমেন্টসমূহ দুই বা ততোধিকবার নির্বাহ এবং শর্ত সাপেক্ষে নির্বাহের প্রয়োজন হয়। এক্ষেত্রে স্টেটমেন্টসমূহের নির্বাহ প্রোগ্রামার কর্তৃক নিয়ন্ত্রিত হয়। যে স্টেটমেন্টের সাহায্যে এরূপ স্টেটমেন্টসমূহের নির্বাহ নিয়ন্ত্রণ করা হয় তাকে কন্ট্রোল স্টেটমেন্ট বলে।

২। লুপ কন্ট্রোল স্টেটমেন্ট কী?

উত্তরঃ সি প্রোগ্রামে কোন স্টেটমেন্ট একাধিকবার নির্বাহের জন্য যে কন্ট্রোল স্টেটমেন্ট ব্যবহৃত হয় তাকে লুপ কন্ট্রোল স্টেটমেন্ট বলে।

অনুধাবনমূলক প্রশ্নসমূহ

১। কখন switch স্টেটমেন্ট ব্যবহার করা উত্তম?