সহজ ভাষায় প্রোগ্রামিং সি/সি++

আহমদ ওয়াহিদ মাসুদ

মানুষের প্রয়োজন ও বিভিন্ন সমস্যার সহজ সমাধানের জন্য স্টেটমেন্ট নিয়ন্ত্রণ এবং লুপিং অনেক গুরুত্বপূর্ণ ভূমিকা পালন করে। স্টেটমেন্ট নিয়ন্ত্রণ সম্পর্কে আগেই আলোচনা করা হয়েছে। তাই এ ব্যাপারে আর বিস্তারিত আলোচনা করা হলো না। লক্ষণীয়, স্টেটমেন্ট নিয়ন্ত্রণ বলতে সাধারণত বিভিন্ন কন্ডিশনাল কাজ বোঝানো হয়। আর লুপিংয়ের মাধ্যমে একই কাজের পুনরাবৃত্তি করা যায়। ইতঃপূর্বে if-else কমান্ড নিয়ে আলোচনা করা হয়েছে। if-else-এর তিন ধরনের ব্যবহার নিয়েও বিস্তারিত আলোচনা করা হয়েছে। তাই এ লেখায় if-else ছাড়া অন্য কী কী কমান্ডের মাধ্যমে লুপিং করা সম্ভব এবং কখন কোন কমান্ড ব্যবহার করা দরকার, তা নিয়ে আলোচনা করাসহ if-else স্টেটমেন্ট নিয়েও কিছু আলোচনা করা হয়েছে।

কখনো কখনো এমন কিছু অবস্থার সৃষ্টি হতে পারে, একটি শর্ত সত্য হলে কোনো কাজ অপর কোনো শর্তের ওপর নির্ভর করবে এবং আগের শর্তটি মিথ্যা হলে কাজটি অন্য কতগুলো শর্তের ওপর নির্ভর করবে। যেমন– রহিম যদি এসএসসি পরীক্ষায় পাস করে থাকে তাহলে সে কলেজে পড়তে পারবে, আর যদি ফেল করে তাহলে সে কলেজে পড়তে পারবে না। আবার রহিম যদি কলেজে পড়তে পারে তাহলে তার জিপিএ চেক করে দেখা হবে, যদি সে ৩.৫-এর ওপরে পায় তাহলে সে সায়েন্সে পড়তে পারবে, অন্যথায় পারবে না। এধরনের সমস্যা সমাধানের জন্য nested if else ব্যবহার করা হয়। অর্থাৎ একটি if-else-এর মধ্যে আরেকটি if-else থাকতে পারে, একাধিকও থাকতে পারে। একটি প্রোগ্রাম উদাহরণ হিসেবে দেয়া হলো :

```
int score=100;
char grade='A';
if(score>=90)
{
        if(grade=='A')
                printf("grade is A");
        else
                printf("grade is not
A");
}
else
        printf("disqualified!");
```

এখানে নেস্টেড if-else ব্যবহার করা হয়েছে। প্রথমে outer if-এর শর্ত যদি সত্য হয় অর্থাৎ স্কোরের মান যদি ৯০-এর সমান অথবা বেশি হয়, তাহলে প্রোগ্রাম inner if-এ বা ভেতরের if-এ ঢুকবে, আর যদি মিথ্যা হয় তাহলে প্রিন্ট করবে 'disqualified!'। প্রোগ্রাম যদি inner if-এ ঢুকে, তাহলে চেক করবে যে গ্রেডের মান A কি না, যদি A হয় তাহলে প্রিন্ট করবে 'grade is A'। আর যদি না হয় তাহলে প্রিন্ট করবে 'grade is not A'।

সুইচ কেস

এবারে যে কমান্ডের ব্যবহার দেখানো হয়েছে, তাকে বলে switch কেস। এটি if-এর মতো একটি কন্ডিশনাল অপারেটর। এর ব্যবহারও if-এর মতো। তাই বিস্তারিত আলোচনা না করে সরাসরি একটি উদাহরণ এবং তার ব্যাখ্যা নিচে দেয়া হলো।

```
int grade=8;
switch(grade)
{
        case 10:
        case 9:
                printf("grade=9");
                break;
        case 8:
                printf("grade=8");
                break;
        case 7:
                printf("grade=7");
                break;
default:
                printf("no score!");
                break;
}
```

উপরের উদাহরণ দেখে হয়ত কিছুটা ধারণা পাবেন, কিভাবে সুইচ কেস কাজ করে। ধারণা না পেলেও অন্তত সিন্টেক্স বোঝা যাচ্ছে। সুইচের পাশে প্রথম বন্ধনীর ভেতরে এক্সপ্রেশন থাকে। সেই এক্সপ্রেশনের মান যদি নন-জিরো (অশূন্য) হয় তাহলে প্রোগ্রাম কেসটির ভেতরে ঢুকবে, অন্যথায় ঢুকবে না। এখন সেই এক্সপ্রেশনটি যেই মান রিটার্ন করবে প্রোগ্রাম তত নম্বর কেসে প্রবেশ করবে। যেমন– এখানে এক্সপ্রেশন হিসেবে শুধু grade ব্যবহার করা হয়েছে। তার মানে হলো, গ্রেডের মান যত হবে প্রোগ্রাম তত নাম্বার কেসে প্রবেশ করবে। আর যদি গ্রেডের মান শূন্য হয়, তাহলে প্রোগ্রাম এই সুইচ কেসে প্রবেশই করবে না। এখানে গ্রেডের মান হলো ৮। তাই প্রোগ্রাম ৮ নাম্বার কেসে প্রবেশ করবে। এরপর উল্লিখিত লাইনটি প্রিন্ট করবে। এরপর ব্রেক করবে অর্থাৎ প্রোগ্রাম এই কেস থেকে বের হয়ে যাবে, কিন্তু বন্ধ হয়ে যাবে না, কেসের পরবর্তী কাজ সম্পাদন করবে। সুতরাং ব্রেকের কাজ হলো প্রোগ্রাম যেই লুপ/কেসে আছে সেখান থেকে বের হয়ে যাওয়া। তবে if-else কোনো লুপ/কেস নয় সে ব্যাপারে সবসময় খেয়াল রাখতে হবে। এখন কথা হচ্ছে উপরের কোডে প্রতিটি কেসের পর যদি ব্রেক দেয়া না থাকত তাহলে কি হতো? এর উত্তর হলো, কেসের ভেতরে যদি ব্রেক না থাকত তাহলে প্রোগ্রাম সংশ্লিষ্ট কেসের কাজ সম্পাদন করত এবং তারপর পরবর্তী কেসগুলোর কাজও সম্পাদন করত। এখানে কেস ১০-এ কিছু লেখা নেই, সুতরাং প্রোগ্রাম যদি কেস ১০-এ প্রবেশ করে তাহলে তা কোনো কমান্ড পাবে না, তাই প্রোগ্রাম পরবর্তী কেসে প্রবেশ করবে। প্রতিটি সুইচ কেসেই একটি করে ডিফল্ট কেস দেয়া থাকে, উপরের কোডটিতেও দেয়া আছে। এর কারণ, যদি কখনো এমন কোনো অবস্থার সৃষ্টি হয় যে গ্রেডের মান ৭ থেকে ১০-এর মধ্যে নেই অর্থাৎ তা কোনো কেসেই পড়ছে না তখন প্রোগ্রাম ডিফল্ট কেসে প্রবেশ করবে। আরেকটি বিষয় সবসময় খেয়াল রাখতে হবে– যেকোনো কেসের ইনডেক্সিংয়ের জন্য (অর্থাৎ কেস ১০, কেস ৯, কেস ৮ ইত্যাদি নামকরণ) সবসময় কোনো মান ব্যবহার করতে হবে। যেমন– ৯, ১০, ৮ ইত্যাদি মান, বিভিন্ন ক্যারেক্টার যেমন a, A, b, c, D, d ইত্যাদি ব্যবহার করা যাবে। আমরা জানি, প্রতিটি ক্যারেক্টারের একটি করে আসকি মান (ASCII value) আছে, যা দিয়ে প্রোগ্রাম ওই ক্যারেক্টারকে শনাক্ত করতে পারে। তাই যদি কেস 'a', কেস 'b' ইত্যাদি নামকরণ করা হয়, তাহলে প্রকৃতপক্ষে ওই ক্যারেক্টারের আসকি মানগুলো ব্যবহার হয়। তবে ইউজারের সেটি নিয়ে চিন্তার কোনো দরকার নেই, কারণ তিনি ওইসব ক্যারেক্টার দিয়েই প্রোগ্রামিং করতে পারবেন এবং তাতে কোনো সমস্যা হবে না। সুইচ কেসের মধ্যে নেস্টেড সুইচ কেসও ব্যবহার করা যায়।

এবারে লুপ নিয়ে আলোচনা করা যাক। লুপিংয়ের মূল উদ্দেশ্য, অল্প কিছু ইনস্ট্রাকশনের মাধ্যমে একই কাজ প্রয়োজনানুসারে বারবার সম্পাদন করা।

while loop

ছোট একটি প্রোগ্রাম উদাহরণ হিসেবে দেয়া হলো এবং তার ভিত্তিতে এই লুপটির বিস্তারিত আলোচনা করা হয়েছে নিচে।

```
int i=1;
while(1)
{
        printf("%d ",i);
        i++;
if(i==100)
        break;
}
```

While লুপের সিনটেক্স একেবারেই সহজ। While কিওয়ার্ডটি লিখে তার পাশে এক্সপ্রেশন দিতে হয়। এক্সপ্রেশনের মান যদি নন-জিরো হয় তাহলে লুপটির অধীনে যে কোড ব্লকটি আছে, তা সম্পাদন করা হয়। তবে কন্ডিশন বা এক্সপ্রেশনের মান যতক্ষণ সত্য হবে লুপটি ততক্ষণ চলতে থাকবে। উপরের উদাহরণটি

(বাকি অংশ ৮৭ পৃষ্ঠায়)

ব্যাকআপ তৈরি করা

কেউ বলতে পারবে না কখন, কিভাবে আপনার ফাইল করাপ্ট করবে। তবে মনে রাখা দরকার, সফটওয়্যার ম্যালফাংশন বা হার্ডওয়্যারের ড্যামেজের কারণে ফাইল করাপ্ট করতে পারে। তাই ডাটার নিরাপত্তার কথা বিবেচনায় রেখে ডাটার ব্যাকআপ তৈরি করা উচিত। ডাটা ব্যাকআপ করার জন্য ক্লাউড সার্ভার অপশন যেমন বেছে নিতে পারেন, তেমনি ভাবতে পারেন বিভিন্ন ড্রাইভে ফাইলের মাল্টিপল কপি সেভ করার কথা। বিপুল আকারের ডাটা ব্যাকআপের জন্য দরকার হতে পারে কিছু বিশেষ ধরনের সফটওয়্যার। সুপ্রতিষ্ঠিত ম্যানুফ্যাকচারার যেমন এইচপি, ডেল, লেনোভো প্রভৃতি কোম্পানি তাদের পণ্যের সাথে সিকিউরিটি স্যুট দিয়ে থাকে যেখানে সম্পৃক্ত থাকে ব্যাকআপ ফিচারও। যদি এ ধরনের কোনো স্যুট না থাকে তাহলে ব্যবহার করতে পারেন EaseUS Todo Backup বা Cobian Backup ধরনের সফটওয়্যার। এসব সফটওয়্যারের সাথে নির্দিষ্ট ফোল্ডার বা পুরো ড্রাইভের ব্যাকআপ তৈরির অপশন সমন্বিত রয়েছে। কিছু ব্যাকআপ সফটওয়্যার অফার করে ডাটা এনক্রিপ্ট অপশন এবং ব্যাকআপ ড্রাইভ বা ফোল্ডারের সাইজ কমানোর জন্য কম্প্রেস ফাইলও অফার করে।

ভালো সিকিউরিটি স্যুট

ভাইরাস বা ম্যালওয়্যার থেকে রক্ষা পাওয়ার সবচেয়ে ভালো ও বিশ্বস্ত উপায় হলো ভালো ও কার্যকর অ্যান্টিভাইরাস ব্যবহার করা। ইদানীং বেশিরভাগ সিকিউরিটি স্যুটের সাথে থাকে বান্ডেল আকারে বিভিন্ন ধরনের সার্ভিস, যেমন অ্যান্টিভাইরাস প্রোটেকশন থেকে শুরু করে স্প্যাম ব্লকার। মনে রাখা দরকার, একটি ভালো সিকিউরিটি সফটওয়্যার আপনাকে বিভিন্ন উপায়ে সুরক্ষিত করবে। যদি আপনার প্রতিষ্ঠানটি বেশ বড় ধরনের হয় যেখানে ডাটার নিরাপত্তার বিষয়টি খুবই গুরুত্বপূর্ণ সেখানে ফ্রি ভার্সনের সিকিউরিটি টুল ব্যবহার করার পরিবর্তে পেইড ভার্সনের টুল ব্যবহার করা উচিত। কেননা ফ্রি ভার্সনের টুলগুলো কিছু দিনের জন্য ফ্রি থাকে, যা ট্রায়াল ভার্সন হিসেবে পরিচিত। এসব ফ্রি ভার্সনে আবার কিছু কিছু ফিচার ব্লক করা থাকে। এ ছাড়া আরেকটি গুরুত্বপূর্ণ বিষয় হলো আপনার ব্যবহারের সফটওয়্যারগুলো যেনো সবসময় আপডেটেড হয় সে ব্যাপারে নিশ্চিত থাকতে হবে। ভালো মানের অ্যান্টিভাইরাস প্রোগ্রামগুলো নিয়মিতভাবে আপডেট করা হয় সর্বশেষ ভাইরাস ডাটাবেজ এবং লুপহোলের জন্য ফিক্স দিয়ে। সুতরাং নিয়মিত আপডেট দিয়ে পিসি কনফিগার করা খুবই গুরুত্বপূর্ণ কাজ।

ডাটা এনক্রিপ্ট করা

যদি আপনার ডাটা খুবই গুরুত্বপূর্ণ ও সংবেদনশীল হয়ে থাকে তাহলে এসব মূল্যবান ডাটা ইন্টারনেট বা এক্সটারনাল স্টোরেজ মিডিয়ার মাধ্যমে ট্রান্সফার করার সময় আপনাকে নিশ্চিত করতে হবে যে এসব ডাটায় থার্ড পার্টি ব্যবহারকারীরা যাতে অ্যাক্সেস না পায়। কোনো কোনো ক্ষেত্রে ডাটার সুরক্ষার জন্য ব্যবহার করা উচিত এনক্রিপশন সফটওয়্যার।

মাইক্রোসফট উইন্ডোজ ৭-এর কোনো কোনো ভার্সনে বিটলকার (BitLocker) টুলকে বান্ডেল আকারে দিয়েছে ব্যবহারকারীদেরকে। এই টুল ব্যবহার করে ব্যবহারকারীরা তাদের পিসি ও রিমুভাল ড্রাইভের ডাটাকে এনক্রিপ্টকরতে পারবে। এই সফটওয়্যারটি উইন্ডোজ ৭ আল্টিমেট বা এন্টারপ্রাইজ ভার্সনেই পাওয়া যাবে। তবে এনক্রিপ্টেড ফাইল ভিস্তা এবং এক্সপি অপারেটিং সিস্টেমেও পড়া যাবে।

ইচ্ছে করলে আপনি ট্রুক্রিপট (TrueCrypt) নামের ফ্রি টুল ব্যবহার করতে পারবেন। এই টুলটি মাল্টিপ্লাটফরম ওপেন সোর্স এনক্রিপশন প্রোগ্রাম ট্রুক্রিপট টুল দিয়ে ফাইল এনক্রিপটকরা যায় ড্রাইভে ফোল্ডার তৈরি করার মাধ্যমে এবং গুরুত্বপূর্ণ ও সংবেদনশীল ডকুমেন্টকে এনক্রিপ্টেড ফোল্ডারের মধ্যে রেখে অথবা সম্পূর্ণ পার্টিশন বা স্টোরেজ ডিভাইস যেমন ইউএসবি ফ্ল্যাশড্রাইভ বা পোর্টেবল হার্ডডিস্ককে এনক্রিপটকরতে পারবেন। ট্রুক্রিপট প্রোগ্রাম ব্যবহার করে মাল্টিপল এনক্রিপশন অ্যালগরিদম এবং স্টোর করে এনক্রিপশন ফাইলকে স্টোরেজ ডিভাইসের ট্রুক্রিপট এনক্রিপটেড ভলিউমে যা সাধারণ ফোল্ডারের মতো এবং এনক্রিপটেড ফাইল শনাক্ত করাকে প্রায় অসম্ভব করে ফেলেছে।

হার্ডওয়্যার নিরাপদ রাখা

ডাটা সুরক্ষার ব্যাপারে নিশ্চিত থাকতে চাইলে স্টোরেজ হার্ডওয়্যারকে অবশ্যই নিরাপদ রাখতে হবে। হার্ডওয়্যারের ফিজিক্যাল ড্যামেজ খুব সহজেই হতে পারে। তাই এক্সটারনাল স্টোরেজ ডিভাইসকে নিরাপদ জায়গায় যেমন রাখতে হবে তেমনি নিশ্চিত করতে হবে শক প্রুফকে। তাছাড়া ভোল্টেজ স্ট্যাবিলাইজার এবং ইউপিএস ব্যবহার করা উচিত বাধ্যতামূলকভাবে। কেননা আমাদের দেশে বিদ্যুতের বেহাল দশার কারণ ভোল্টেজের ওঠানামা যেমন হয় খুব বেশি তেমনি হয় বিদ্যুৎ যখন-তখন আসা-যাওয়া। এ দুটো বিষয়ই মূলত আমাদের দেশে হার্ডড্রাইভ ড্যামেজের প্রধান কারণ, যার কারণে লোকাল ডাটা তথা অফলাইন ডাটা হারানোর ঘটনা বেশি ঘটে থাকে।

ফিডব্যাক : swapan52002@yahoo.com

প্রোগ্রামিং সি/সি++

(৮৫ পৃষ্ঠার পর)

দেখলে বোঝা যাবে, লুপটি একটি অসীম লুপ। কারণ, এই লুপের এক্সপ্রেশন হিসেবে সরাসরি একটি নন-জিরো মান দেয়া হয়েছে এবং এই মানটির পরিবর্তনের কোনো ইনস্ট্রাকশন দেয়া নেই। অর্থাৎ যতবার এই এক্সপ্রেশনের মান চেক করা হবে ততবার একটি নন-জিরো মান পাওয়া যাবে অর্থাৎ লুপের কন্ডিশন সত্য বলে বিবেচিত হবে। প্রথমবার প্রোগ্রাম যখন লুপটিতে আসবে, তখন দেখবে যে কন্ডিশন সত্য, সুতরাং প্রোগ্রাম লুপে প্রবেশ করবে। এরপরে i-এর মান (১) প্রিন্ট করবে। এরপর i-এর মান ১ বাড়বে। তারপর চেক করবে i-এর মান ১০০ কি না। যদি ১০০ হয়, তাহলে লুপটি ব্রেক করবে অর্থাৎ প্রোগ্রাম লুপ থেকে বের হবে। আর যদি ১০০ না হয়, তাহলে প্রোগ্রাম আবার লুপের কন্ডিশনে ফিরে যাবে এবং দেখবে যে কন্ডিশন কি সত্য না মিথ্যা। এবারেও কন্ডিশন সত্য হবে, কারণ এক্সপ্রেশন হিসেবে সরাসরি একটি নন-জিরো মান দেয়ায় লুপটি অসীম হয়ে গেছে। দ্বিতীয়বার প্রোগ্রাম লুপটিতে প্রবেশ করার পর আবার i-এর মান প্রিন্ট করবে। এবার কিন্তু i-এর মান হিসেবে ২ প্রিন্ট করবে, কারণ পূর্বের লুপে i-এর মান ১ ইনক্রিমেন্ট অর্থাৎ বাড়ানো হয়েছে। এভাবে লুপটি চলতে থাকবে। অর্থাৎ এই প্রোগ্রামটি চালালে ১ থেকে ১০০ পর্যন্ত প্রিন্ট করা হবে। ১০০ প্রিন্ট করার পর যখন i-এর মান ১০১ হয়ে যাবে তখন প্রোগ্রাম Bd স্টেটমেন্টে প্রবেশ করবে এবং তখন if-এর কন্ডিশন সত্য হওয়ার জন্য প্রোগ্রাম লুপটি ব্রেক করবে। উল্লেখ্য, প্রোগ্রাম যখন একবার কোনো লুপ বা কেস ব্রেক করে তখন তাকে নির্দেশ না দেয়া পর্যন্ত নিজে থেকে কখনও ওই লুপ/কেসে প্রবেশ করে না।

কয়েক ধরনের এমন লুপ আছে সি ল্যাঙ্গুয়েজে। এদের একেকটির কাজ একেক রকম। তবে একটির কাজ অন্যটি দিয়ে করা সম্ভব। কেস এবং লুপ সম্পর্কে আরও ভালোভাবে জানার জন্য ইন্টারনেট ব্যবহার করা প্রয়োজন। ইন্টারনেট থেকে বিভিন্ন টিউটোরিয়াল দেখলে লুপ/কেস সম্পর্কে ধারণা আরও স্পষ্ট হবে।

ফিডব্যাক : wahid_cseaust@yahoo.com

www.comjagat.com

'কমজগৎ ডট কম' বাংলা ভাষায় সবচেয়ে বড় ও তথ্যসমৃদ্ধ ওয়েব পোর্টাল। এতে মাসিক কমপিউটার জগৎ-এ প্রকাশিত সব তথ্য অন্তর্ভুক্ত করা হয়েছে। এটি বাংলাদেশে তথ্যপ্রযুক্তিভিত্তিক প্রথম ও বহুল প্রচারিত মাসিক পত্রিকা, যা ১৯৯১ সালের মে মাস থেকে নিয়মিতভাবে প্রকাশিত হয়ে আসছে।