

প্রোগ্রামিং ল্যান্ডুয়েজ ব্যবহার করা হয় বিভিন্ন সফটওয়্যার বানানোর জন্য। আর সি যেহেতু একটি স্ট্যান্ডার্ড ল্যান্ডুয়েজ, তাই এটি দিয়েও সফটওয়্যার বানানো সম্ভব। এ পর্বে স্ট্রাকচার ছাড়া অন্যান্য কাস্টম ডাটা টাইপ নিয়ে আলোচনা করা হয়েছে। একই সাথে সি-তে কীভাবে ফাইল ওপেন করা যায় তাও দেখানো হয়েছে।

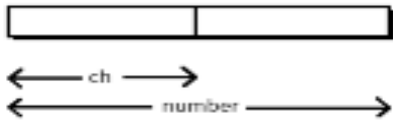
ইউনিয়ন

স্ট্রাক্ট কিওয়ার্ডের মতো ইউনিয়নকেও সি-তে কাস্টম ডাটা টাইপে তৈরি করে ব্যবহার করা যায়। ইউনিয়ন ডিক্লেয়ার করার সিনট্যাক্স হলো

```
union x
{
    char ch;
    int num-
    ber;
}x1, x2;
```

উপরের উদাহরণ

থেকে দেখা যাচ্ছে, স্ট্রাক্টের মতোই ইউনিয়নকে ডিক্লেয়ার করতে হয়। এ ছাড়া ডিক্লেয়ার করার পর একই সাথে ভেরিয়েবলকেও ডিক্লেয়ার করা যায়। স্ট্রাকচার যেভাবে মেমরিতে জায়গা দখল করে, ইউনিয়নও একইভাবে করে। পার্থক্য হচ্ছে ইউনিয়নের প্রতিটি মেম্বারের জায়গা শুরু থেকে অ্যালোকেট করা হবে। সুতরাং উপরের ইউনিয়ন যদি একটি স্ট্রাকচার হতো, তাহলে তা মেমরিতে ৩ বাইট জায়গা নিত, কিন্তু ইউনিয়নের ক্ষেত্রে তা ২ বাইট নেবে।



চিত্রঃ মেমরিতে ইনামের গঠন

চিত্রে দেখানো হয়েছে কীভাবে মেমরিতে ইউনিয়ন ভেরিয়েবল জায়গা অ্যালোকেট করে।

ব্যবহারের ক্ষেত্রে স্ট্রাকচারের সাথে ইউনিয়নের কিছুটা পার্থক্য আছে। প্রোগ্রামে অনেক সময় এমন হতে পারে, একই কাজের জন্য একাধিক ভেরিয়েবলের দরকার ঠিকই, কিন্তু সব ভেরিয়েবল কখনও একসাথে ব্যবহার হবে না। এ ধরনের ক্ষেত্রে ইউনিয়ন ব্যবহার হয়, কেননা ইউনিয়নের জন্য মেমরিতে কম জায়গা অ্যালোকেট হবে।

স্ট্রাকচারের বিভিন্ন মেম্বারের জন্য যেভাবে মান নির্ধারণ করা হয়, ইউনিয়নের জন্য একই নিয়মে মান নির্ধারণ করা হয়। অর্থাৎ এক্ষেত্রেও মেম্বার অপারেটর অথবা অ্যারো অপারেটর ব্যবহার করতে হয়। তবে পার্থক্য হলো, ইউনিয়নে কোনো মেম্বারের জন্য নতুন ডাটা নির্ধারণ করা হলে আগের ডাটা আর থাকে না। ইউনিয়ন ব্যবহারের সময় বিষয়টি অবশ্যই খেয়াল রাখতে হবে, অন্যথায় প্রোগ্রামে

অপ্রত্যাশিত ফলাফল আসতে পারে।

ইনিউমারেশন

সি-তে স্ট্রাকচার কিংবা ইউনিয়নের মতো ইনিউমারেশন পদ্ধতিতেও কাস্টম ডাটা তৈরি করা যায়। তবে ইনামের (ইনিউমারেশনকে সংক্ষেপে ইনাম বলা হয়) মেম্বার ব্যবহার করা এবং এর কাজ করার পদ্ধতিটি একটু ভিন্ন ধরনের।

ইনামের ডাটা টাইপ ডিক্লেয়ার করার নিয়ম হলো :

```
enum tag
{
    enum list or
    enum member(s);
}
```

এক বা একাধিক কনস্ট্যান্ট। এই মান একজন প্রোগ্রামার যেমন নির্ধারণ করতে পারেন, তেমনি কম্পাইলার নিজেও নির্ধারণ করতে পারে। মেম্বারের মান কম্পাইলার নির্ধারণ করলে প্রথম মেম্বারের মান হয় ০, পরেরটির মান হয় ১। এভাবে মেম্বারের মান ১ করে বাড়তে থাকে।

টাইপডেফ

সি-তে কাস্টম ডাটা টাইপে তৈরির আরেকটি পদ্ধতি হলো টাইপডেফ ব্যবহার করা। তবে টাইপডেফের মাধ্যমে আসলে কোনো নতুন ডাটা টাইপ তৈরি হয় না, বরং এর মাধ্যমে বিল্টইন কিংবা অন্য কাস্টম ডাটা টাইপের জন্য নতুন নাম নির্ধারণ করা হয়। সেই হিসেবে একে কাস্টম ডাটা টাইপ বলা যেতে পারে। টাইপডেফ ব্যবহারের সিনট্যাক্স হলো :

```
typedef data_type
new_name;
```

এখানে ডাটা টাইপ হলো যেকোনো বিল্টইন অথবা অন্য কাস্টম

ডাটা টাইপের নাম এবং নিউ নেম হলো নির্ধারিত নতুন ডাটা টাইপের নাম। যেমন : প্রোগ্রামে যদি নিচের মতো করে কোড লেখা হয়,

```
typedef char word;
```

তাহলে ওয়ার্ড হবে ক্যারেক্টার টাইপের জন্য নির্ধারিত নতুন নাম। এরপর থেকে যেকোনো ক্যারেক্টার ভেরিয়েবল ডিক্লেয়ার করার জন্য ভেরিয়েবলের নামের আগে char না লিখে ওয়ার্ড লিখলেই হবে, অথবা ইউজার চাইলে পুরনো নামও ব্যবহার করতে পারেন। ইনামের ক্ষেত্রে টাইপডেফের একটি উদাহরণ দেয়া যেতে পারে। আগেই বলা হয়েছে ইনাম লেখার নিয়ম হলো :

```
enum day {sat, sun, mon};
enum day today, nextday;
```

কিন্তু এখানে টাইপডেফ ব্যবহার করে আরও সহজে ডিক্লেয়ার করা যায়। যেমন :

```
enum day {sat, sun, mon};
typedef enum day day;
day today, nextday;
```

আপাতত এটি অতিরিক্ত কোড মনে হলেও বড় প্রোগ্রামের ক্ষেত্রে এটি অনেক সাহায্য করে। যেমন : এমন অনেক পরিস্থিতির সৃষ্টি হতে পারে, যেখানে প্রোগ্রামের শুরুতে স্ট্রাকচার ডিক্লেয়ার করতে হবে, কিন্তু কোডের একেক জায়গায় ওই স্ট্রাকচারের ভেরিয়েবল ডিক্লেয়ার করতে হবে। তখন বারবার স্ট্রাকচার কিওয়ার্ড না লিখে টাইপডেফ দিয়ে তার জন্য নিজের পছন্দমতো একটি নাম সেট করে নিয়ে ওই নামটিকে ব্যবহার করলেই হবে। এতে প্রোগ্রামারের কষ্ট অনেকাংশেই কমে। এ ছাড়া টাইপডেফের মাধ্যমে যদি সহজ কোনো নাম দেয়া হয়, তাহলে পরবর্তী সময়ে তা ব্যবহার করার সময় প্রোগ্রামার নামটি সহজে মনে করতে

সহজ ভাষায় প্রোগ্রামিং সি/সি++

আহমদ ওয়াহিদ মাসুদ

এখানে ইনাম কিওয়ার্ড ব্যবহার করে কম্পাইলারকে জানিয়ে দেয়া হয় ডাটা টাইপটি একটি ইনাম টাইপের ডাটা। স্ট্রাকচার কিংবা ইউনিয়নের ট্যাগের মতো ইনামের ট্যাগের কাজও একই। ইনাম ডাটা টাইপের ক্ষেত্রে মেম্বার হলো এক বা একাধিক ইন্টিজার টাইপের কনস্ট্যান্টের তালিকা, যাদের প্রত্যেকের একটি নির্দিষ্ট মান থাকে। এ ক্ষেত্রে প্রতিটি মেম্বারকে কমা দিয়ে লেখা হয়। যেমন :

```
enum day (sat, sun, mon, tue, wed, thu, fri);
```

এখানে শুধু একটি ইনাম টাইপ, যার মেম্বার হিসেবে বিভিন্ন দিনের নাম ব্যবহার করা হয়েছে। প্রোগ্রামে স্ট্রাকচার ভেরিয়েবলের মতোই ইনাম ডাটা টাইপের ভেরিয়েবল ডিক্লেয়ার করা হয়। যেমন : উপরে একটি ইনাম টাইপ ডাটা ডিক্লেয়ার করা হয়েছে। এখন ইউজার যদি এই টাইপের ভেরিয়েবল ডিক্লেয়ার করতে চান, তাহলে নিচের মতো করে কোড লিখতে হবে :

```
enum day week_day;
```

তবে ভেরিয়েবল পরে ডিক্লেয়ার না করে ডাটা টাইপের সাথেও ডিক্লেয়ার করা যায়। যেমন :

```
enum day (sat, sun, mon, tue, wed, thu, fri)
week_day;
```

ইনাম ভেরিয়েবলের মান নির্ধারণের পদ্ধতি স্ট্রাকচার বা ইউনিয়ন থেকে কিছুটা ভিন্ন। নিচে একটি উদাহরণ দেয়া হলো :

```
enum_variable=enum_member;
```

অর্থাৎ ইনাম মেম্বারকেই ইনাম ভেরিয়েবল হিসেবে নির্ধারণ করতে হয়। স্ট্রাকচার কিংবা ইউনিয়ন মেম্বারের মতো ইনাম মেম্বার কোনো ভিন্ন ডাটা টাইপের নয়। বরং ইনাম মেম্বার হলো

পারেন, বারবার অরিজিনাল নামটি খুঁজে বের করে দেখতে হয় না।

সি-তে ডাটা ফাইল

কমপিউটারে বিভিন্ন কাজ করার জন্য অনেক ধরনের সফটওয়্যার ব্যবহার করতে হয়। এসব সফটওয়্যারের এক বা একাধিক .exe, .com, .bat, .txt, .doc ইত্যাদি এক্সটেনশনবিশিষ্ট প্রয়োজনীয় ফাইল থাকে। তবে যখন কোনো প্রোগ্রাম চালাতে হয়, তখন শুধু সংশ্লিষ্ট .exe বা .com ফাইলগুলোই রান করে। ইএক্সই ফাইলের পুরো নাম এক্সিকিউটেবল ফাইল। এটি উইন্ডোজের জন্য ব্যবহার হয়। আর লিনআক্স বা অন্যান্য অপারেটিং সিস্টেমের এক্সিকিউটেবল ফাইলের এক্সটেনশন ভিন্ন ধরনের। এর অর্থ শুধু উইন্ডোজে ইএক্সই ফাইল রান করবে, অন্যান্য অপারেটিং সিস্টেমে নয়।

আবার সফটওয়্যার চালিয়ে ইউজার নতুন কোনো ফাইল তৈরিও করতে পারেন। যেমন : মাইক্রোসফট ওয়ার্ড ব্যবহার করে ডক ফাইল তৈরি করা যায়। এসব ফাইলের কাজ হলো ডাটা সংরক্ষণ করা। তাই এ ধরনের ফাইলকে বলা হয় ডাটা ফাইল। প্রায় প্রতিটি প্রোগ্রামিং ল্যাঙ্গুয়েজেই ডাটা ফাইল তৈরির ব্যবস্থা রয়েছে। এখন দেখানো হয়েছে সি-তে কীভাবে ডাটা ফাইল তৈরি করা এবং তা ব্যবহার করা যায়। তবে তার আগে ডাটা ফাইলের প্রয়োজনীয়তা সম্পর্কে বলে নেয়া দরকার।

প্রোগ্রামে ভেরিয়েবলের মাধ্যমে যেসব ডাটা নিয়ে কাজ করা হয় তা শুধু প্রোগ্রাম চলার সময়ই মেমরিতে থাকে। প্রোগ্রাম বন্ধ হয়ে গেলে ডাটাগুলোও মেমরি থেকে মুছে ফেলা হয়। তাই প্রোগ্রাম চালিয়ে আগের ডাটাগুলো যদি আবার আনতে হয়, তাহলে ডাটাগুলোকে নতুন মেমরিতে অ্যাসাইন করতে হয়, যা অনেক কষ্টসাধ্য ব্যাপার। একজন ইউজার যদি আগের টাইপ করা ডাটা আবার ব্যবহার করতে চান, তাহলে প্রোগ্রামে ডাটা ফাইল তৈরির ব্যবস্থা থাকতে হবে, যাতে ইউজার তার সেভ করা ডাটাগুলোকে হার্ডডিস্কে রেখে দিতে পারেন এবং এ কাজটি শুধু ডাটা ফাইলের মাধ্যমেই করা সম্ভব।

সি-তে নতুন কোনো ফাইল তৈরি করতে বা আগের তৈরি করা কোনো ফাইলকে ব্যবহার করতে FILE নামের স্ট্রাকচার ব্যবহার করতে হয়। আর ফাইলের বিভিন্ন অপারেশনের জন্য আলাদা বিল্টইন ফাংশন আছে। যেমন : একটি ফাইল ওপেন করতে fopen() ফাংশন ব্যবহার করতে হয়। নিচে ফাইল ওপেন করার সিনট্যাক্স দেয়া হলো :

```
Header file: stdio.h
file* file_pointer;
fopen("file_name", "mode");
```

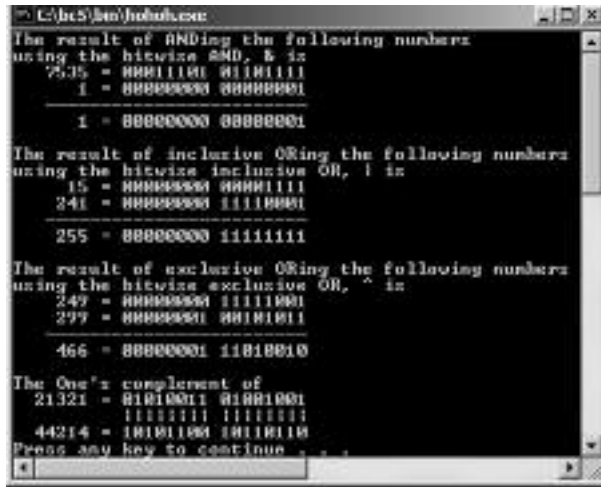
এখানে FILE হলো একটি স্ট্রাকচার, যেটি

উপরিষ্কারিত হেডার ফাইলে বর্ণিত আছে। আর file_pointer হলো একটি পয়েন্টার ভেরিয়েবলের নাম। এই পয়েন্টার দিয়েই একটি ফাইলের বিভিন্ন অপারেশন করা হয়।

fopen() ফাংশন ব্যবহার করে যেকোনো ফাইলকে শুধু ওপেন করা যাবে, কিন্তু আর কোনো কাজ করা যাবে না। আর যে ফাইলকে ওপেন করা হচ্ছে সে ফাইলকে আগে ডিক্লোর করা পয়েন্টার দিয়ে পয়েন্ট করতে হবে। যেমন :

```
file* fp;
fp=fopen("new.txt", "w");
```

এখানে এফপি নামের একটি ফাইল পয়েন্টার ডিক্লোর করা হয়েছে। এরপর নিউ নামের একটি টেম্পট ফাইল খুলে তাকে এফপি দিয়ে পয়েন্ট করা হচ্ছে। লক্ষণীয়, fopen() ফাংশনের দুটি প্যারামিটার। প্রথমটি হলো ফাইলের নাম এবং পরেরটি হলো মোড। এই ফাংশনে মোট তিনটি মোড আছে। যেমন : r, w, a। অর্থাৎ প্রথম মোড হলো রিড মোড। এই মোডে আগের তৈরি করা কোনো ফাইলকে শুধু ওপেন করে রিড



করা যাবে, কিন্তু ফাইলের ডাটার কোনো পরিবর্তন করা যাবে না। দ্বিতীয় মোড হলো রাইট মোড। এই মোড দুইভাবে কাজ করে। এক. রাইট মোডে, যে নামের ফাইল ওপেন করা হয় সেই নামের ফাইল যদি আগে থেকে না থাকে তাহলে ওই নামে নতুন একটি ফাইল তৈরি হবে। দুই. সেই নামে যদি একটি ফাইল আগে থেকেই তৈরি করা থাকে, তাহলে ওই ফাইলটিই ওপেন হবে, কিন্তু তার ডাটা সব মুছে যাবে। তৃতীয় মোড হলো অ্যাপেন্ড মোড। এই মোডও দুইভাবে কাজ করে। এক. যে নামের ফাইল ওপেন করা হয়, সেই নামের ফাইল যদি আগে থেকে না থাকে, তাহলে ওই নামে নতুন একটি ফাইল তৈরি হবে। দুই. সেই নামে যদি একটি ফাইল আগে থেকেই তৈরি করা থাকে, তাহলে ওই ফাইলটিই ওপেন হবে, কিন্তু তার কোনো ডাটা মুছে যাবে না। বরং তাতে নতুন করে কোনো ডাটা লেখা হলে তা আগের সেভ করা ডাটার পরে যোগ হয়ে যাবে। তবে বেশিরভাগ কম্পাইলারে আরও তিনটি অতিরিক্ত মোড দেখা

যায়। যেমন : a+ মোড অনেকটা অ্যাপেন্ড মোডের মতোই, তবে এটি ব্যবহার করলে ফাইল থেকে কিছু রিড করাও যাবে। r+ মোড ব্যবহার করলে ফাইল থেকে রিড কিংবা রাইট দুটোই করা যাবে। w+ মোডেও ফাইল থেকে রিড কিংবা রাইট দুটোই করা যায়, তবে এ ক্ষেত্রে নির্ধারিত ডিরেক্টরিতে একই নামে কোনো ফাইল আগে থেকে থাকলে তা মুছে যাবে।

ফাইল তৈরি বা ওপেনের ক্ষেত্রে আরেকটি বিষয় লক্ষ করতে হয়। তা হলো ফাইল লোকেশন। ফাংশনের প্রথম প্যারামিটারে যদি শুধু ফাইলের নাম দেয়া হয়, তার মানে ফাইলটির লোকেশন হচ্ছে রুট ডিরেক্টরি। অর্থাৎ যে প্রোগ্রামের মাধ্যমে ফাইল তৈরি করা হচ্ছে, সেই প্রোগ্রামের ইএক্সই ফাইল যেখানে আছে সেই ডিরেকশন। যেমন : ইউজার শুধু ফাইল ওপেন করার একটি প্রোগ্রাম লিখলেন। আমরা জানি সি-তে প্রোগ্রাম কম্পাইল করলে তার একটি ইএক্সই ফাইল তৈরি হয়, যেটি দিয়ে ইউজার পরবর্তী সময়ে কম্পাইলার ছাড়াই ওই প্রোগ্রামটি চালাতে পারেন। এখন ধরা যাক

ইউজার ওই প্রোগ্রামটি F:\ড্রাইভে সেভ করলেন। তার মানে প্রোগ্রামটির ইএক্সই ফাইলটিও এফ ড্রাইভেই সেভ হবে। সুতরাং ওই প্রোগ্রামের জন্য রুট ডিরেক্টরি হলো এফ ড্রাইভ। এখন ইউজার যদি অন্য কোনো লোকেশন ব্যবহার করতে চান তাহলে তা ফাংশনের প্যারামিটারের মধ্যেই বলে দিতে হবে। নিচে ভিন্ন লোকেশনে একটি ফাইল অ্যাপেন্ড মোডে ওপেন করার উদাহরণ দেয়া হলো :

```
fopen("d:\new folder\","a");
```

এখানে ফাইলের লোকেশন হলো ডি ড্রাইভের নিউ ফোল্ডারের ভেতরে। এখানে লক্ষণীয় হলো ব্যাকস্ল্যাশ।

আমরা জানি সি-তে ব্যাকস্ল্যাশ সাধারণত কোনো ক্যারেক্টার স্কিপ করার জন্য ব্যবহার করা হয়। কিন্তু ব্যাকস্ল্যাশকেই যদি কোথাও একটি ক্যারেক্টার হিসেবে ব্যবহার করতে হয়, তাহলে তার আগে একটি অতিরিক্ত ব্যাকস্ল্যাশ দিতে হবে। আর এই লোকেশন যদি সঠিক হয়, তাহলে ফিল পয়েন্টার ওই ফাইলকে পয়েন্ট করবে, যার অর্থ হলো ওই ফাইল পয়েন্টার দিয়ে কোনো অপারেশন করলে তা ওই ফাইলে প্রভাব ফেলবে। আর উল্লিখিত লোকেশন যদি ভুল হয় তাহলে প্রোগ্রাম কোনো এরর দেখাবে না, তবে ফাইল পয়েন্টার নালকে পয়েন্ট করে থাকবে। সুতরাং এরর না দেখানো সত্ত্বেও প্রোগ্রামের ফলাফল অপ্রত্যাশিত হবে।

একটি সফটওয়্যার বানাতে যেসব সুবিধার প্রয়োজন তার মোটামুটি সবকিছুই সি-তে আছে। এর মাঝে ফাইলের ব্যবহার অবশ্যই গুরুত্বপূর্ণ। কারণ ফাইল ছাড়া কোনো সফটওয়্যার সম্পূর্ণ হয় না।

ফিডব্যাক : wahid_cseast@yahoo.com