

# স্ট্রাকচার প্রোগ্রামিং এর মৃত্যু, অবজেক্ট ওরিয়েন্টেড প্রোগ্রামিং -এর উত্থান এবং ক্লিপার প্রোগ্রামিং ল্যাংগুয়েজ

এস. এম. মফিদুল হক

প্রবন্ধের নামকরণ থেকে সহজেই অনুমেয় যে বিষয়বস্তু আর্ভিত হতে Clipper প্রোগ্রামিং ল্যাংগুয়েজ কে ঘিরে। অবশ্যই এ ধারাবাহিক রচনার আয়ের তুলন ধরে ক্লিপার এর বিভিন্ন ভবিষ্যৎ প্রোগ্রামিং ক্ষমতা, সফটওয়্যার পরিষ্কৃষ্টনে এর বিশেষকর ক্ষমতা এবং সর্বাধিক XBASE Dialects-এ অবজেক্ট ওরিয়েন্টেড প্রোগ্রামিং হিসাবে এর সামল্য। কিন্তু বাংলাদেশে ক্লিপার নিয়ে কাজ করেন এমন প্রোগ্রামারদের সংখ্যা নগণ্য। আমরা চাই বাংলাদেশের প্রোগ্রামাররা Clipper 5.01 নিয়ে কাজ করুন এবং এর বিভিন্ন পরিষ্কৃষ্টন সুবিধা ব্যবহার করে অতি সহজে এবং অল্প সময়ে সফটওয়্যার তৈরী করুন। এতে আমাদের উপপাসন ক্ষমতা বৃদ্ধি বোধ হতে পারে। তাই আমাদের এ রচনার আয়োজনসূত্রে থাকবে: প্রোগ্রামিং কনসেপ্ট, এর বিবর্তন, ট্রাকচার, procedural ও modular প্রোগ্রামিং এর বিবিধ দিক। এর সুবিধা-অসুবিধা-সীমাবদ্ধতা, অবজেক্ট ওরিয়েন্টেড প্রোগ্রামিং -এর উত্থান এর সম্ভাবনার বিভিন্ন দিক, সফটওয়্যার পরিষ্কৃষ্টনে OOP-প্রোগ্রামিং, হিসাবে ক্লিপার এর সুবিধাবন্দী ইত্যাদি। বিষয়বস্তুর বিস্তারিত আলোচনায় আমরা পরব্রু করব অসংখ্য উদাহরণ, Program listing, উদাহরণ প্রোগ্রাম ইত্যাদি। আর সকল Program listing লেখা হবে প্রকাশিতব্য Softline. Lib থেকে। Softline. lib হলো Clipper 5.01 এর Add-On Library আর এতে থাকবে কয়েক শত User Defined Functions, যেনে কিছু ইন্টারিটিভ ও meta-objects, এদের ব্যবহারের নিয়মাবলী এবং টেস্ট প্রোগ্রামসমূহ।

সর্বশেষে, আমাদের সাথে পাঠকগণ সম্পৃক্ত হয়ে যা যা পাবেন: প্রচুর UDF সহ একটি Clipper Add-on Library, যা ব্যবহার করে প্রোগ্রামাররা তৈরী করতে পারবেন পঠিশালী এ্যাপ্লিকেশন বিশেষকরভাবে বৃদ্ধি করতে পারবেন নিজেদের প্রোগ্রামিং-এর ক্ষমতা ও দক্ষতা উপলব্ধি করতে পারবেন commercial library-র টুটুইন্টাই, সর্বাধিক শিবতে পারবেন কিভাবে UDF লিখতে হয় এবং নিজস্ব লাইব্রেরী তৈরী করতে হয় এবং সাথে পাবেন Object Oriented Programming -এর সূত্রক ধারণা।

জাহ্নল স্তম্ভ করা যায় আমাদের আলোচনায়। একই পিছনে ফিরে যাই। Niklaus Wirth, যিনি Pascal, Modula সহ বেশ কিছু প্রোগ্রামিং ল্যাংগুয়েজের রচয়িতা, যদিও উপর দিক স্ট্রাকচার প্রোগ্রামিং-এর আধিপত্যকর নন, তবুও একে জনপ্রিয় করে তুলতে এক বিরাট ভূমিকা পালন করেছেন। স্ট্রাকচার প্রোগ্রামিং-এর guiding principle বেশ সহজেই প্রকাশ করা যেতে পারে স্তম্ভ করতে হবে মূল সমস্যা নিয়ে, এবং পর্যায়েক্রমে

সমস্যাকে বিভক্ত করতে হবে এর কম্পোনেন্টস এ (subroutines, modules) যতকম পর্যন্ত এর বিভিন্ন কম্পোনেন্টস সঠিকভাবে কাজ না করে।

দ্রুত বেশ কয়েক দশক ধরে সফটওয়্যার পরিষ্কৃষ্টনে স্ট্রাকচার প্রোগ্রামিং, guiding principle-র ভূমিকা পালন করে আসছে। সফটওয়্যার পরিষ্কৃষ্টনে অনেক সময় শ্রম ও অসহনীয় কষ্ট করার পর আনন্দের নিকট এটা খুবই পরিষ্কৃষ্টন যে খুব বড় এবং জটিল প্রোগ্রাম পরিষ্কৃষ্টনে স্ট্রাকচার প্রোগ্রামিং-এর সূত্রসমূহ খুব একটা কার্যকর নয়। কারণ এর assumption হলো, এ্যাপ্লিকেশন এর প্রতিটি component এর সামান্যতম অংশেরও পর্যাপ্ত বিশ্লেণ স্তরতেই তৈরী করার পর সোর্স কোড লিখতে হতে দিতে হবে। কিন্তু জীবনতো এমনটি নয়। যেমনটি নয় ব্যবসায়-বাণিজ্য, বিজ্ঞান-প্রযুক্তি। এনেকি মনুষ্যের মন। তবে কেন হতে পারে একটি সফটওয়্যার প্রোগ্রামের ?

এর একটি গুণনগণিতিক ছবন হলো: প্রোগ্রামারকে ছাড়া সত্ত্বক হয়ে প্রোগ্রাম ডিজাইন করা উচিত ছিল। খুবই ভালো কথা। কিন্তু এভাবে সত্ত্বক হয়ে প্রোগ্রাম ডিজাইন এর পরও একজন প্রোগ্রামারের মাসের পর মাস শ্রম ব্যয় গণ্য হয়ে গেছে এ রকম উদাহরণ অনেক আছে। আমি নিজে এবং আমার জানা অনেক প্রোগ্রামারগণ এর বাইরে নয়।

তবে কেন ট্রাকচারড প্রোগ্রামিং? শিক্ষার্থী, যারা প্রোগ্রামিং-এ হাতে খড়ি দেনন, তাদের কাছে ট্রাকচারড প্রোগ্রামিং হলো এক invaluable discipline। কারণ স্তম্ভতে ইহা শিক্ষাধীপ্যাকে যথেষ্ট সাহায্য করে decomposition technique ও logical analysis রূপে করতে। তাছাড়া ট্রাকচারড প্রোগ্রামিং ভালোমতে কাজও করে উত্কর্ষ পর্যন্ত এক বা কয়েকটি কম্পোনেন্ট-এ প্রোগ্রামটি বিভক্ত সাধারণতঃ এ ধরণের প্রোগ্রাম-এর উদ্দেশ্য একটাই বা কয়েকটি। উদাহরণ স্বরূপ নাম করা যেতে পারে: Sorting algorithm, matrix-কে Inverse matrix-এ রূপান্তরকরণ, array-based-games, random number generation ইত্যাদি। শিক্ষার পরিধাবের জন্য এগুলো হলো খুসেই মডুল। কিন্তু সফটওয়্যার উদ্দেশ্যে ধরনের সফটওয়্যার পরিষ্কৃষ্টনের জন্য এক কমানকটিও দিতে প্রস্তত নয়।

তাহলে ক্রেতা/সফটওয়্যার ব্যবহারকারী কেন ধরণের সফটওয়্যার-এর জন্য প্রোগ্রামারদের টাকা দিবেন? ব্যাপারটি বোঝার জন্য একটু তলিয়ে দেখা যাবে। ধরন একজন ব্যবহারকারী একটা একাউন্টটি প্যাকেজ চান। এ থেকে কোন অবস্থায় এমন উপসংহার টানা যায় না যে একাউন্টটি প্যাকেজটি ব্যবহারকারীর উদ্দেশ্য পূর্ণ করবে। যখন করি, ব্যবহারকারীর উদ্দেশ্য হলো কিভাবে তাঁর

কোম্পানীর মুনাফা বাড়ানো যায়। সত্যি বলতে গেলে একটা সাধারণ মানের কোন "off the shelf" এ-থ-উটিলি প্যাকেজ-এর মুনাফা বাড়ানোর জন্য আশা কোন মডুল নেই। আর এ জন্যই বিদ্যুৎ-পী কা-টাইনজন্ড এ্যাপ্লিকেশন সফটওয়্যার এর এত কদর এবং প্রোগ্রামারদেরও বিপুল ডায়াদি।

সত্যি বলাতে কি রিয়েল ওয়ার্ল্ড প্রোগ্রাম হলো বেশ জটিল। একটা real world প্রোগ্রামকে variably related tasks এর সমন্বয় বলা যেতে পারে। যেমন একটা একাউন্টটি প্যাকেজ এর বিবিধ উদ্দেশ্য থাকতে পারে: Order Only এ invoicingকে সরলীকরণ, packing slip প্রস্তুতকরণ, স্বয়ংক্রীয় ইনভেন্টরীকে update করণ এবং re-order list প্রস্তুতকরণ, সমন্বয়ত নির্ভুল একাউন্ট সম্পর্কীয় লিষ্ট তৈরীকরণ, যে কোন মুহুর্তে কাশ বা ব্যাল্ড ব্যালেন্স নির্ভুলভাবে দেখানো, ইত্যাদি। আবার এ প্রোগ্রাম একটা লোকাল এরিয়া নেটওয়ার্ক হে বিতিয় ব্যবহারকারীর জন্য বিভিন্ন মডুল বা কম্পোনেন্ট ব্যবহারের উপযোগীকরণ। যেমন বিভিন্ন ব্যবহারকারীর জন্য ভাটা বৈধতা এর বিভিন্ন অ্যাক্সেস রহিতকরণ। কোন কোন একাউন্টস অফিসার যেন করবেন তার প্রধান কর্মকর্তার বেতন বা অন্যান্য তথ্যে অ্যেসেস না করতে পারে এবং এরকম বিবিধ বিধাযাচনী। অবশ্যই, আমরা দেখছি যে, রিয়েল ওয়ার্ল্ড প্রোগ্রাম এ সাধারণতঃ অনেক ব্যবহারকারী বহুবিধ ট্যাক্স এর সংস্করণ বা এদের অংশবিশেষ পারস্বরম্ব করে থাকেন। আর রিয়েল ওয়ার্ল্ড প্রোগ্রাম বহুবিধ কাজ সমাধা করতে পারে।

কোন জটিল প্রোগ্রামকে যদি সত্যি সত্যিই পরিষ্কারভাবে ডিফাইন করা যায় এবং এক বা একাধিক কমপিউটার বা মডুল-এ সমাধান করা যায়, তবে কিছুদিন পরেই দেখা যাবে যে প্রোগ্রামটি আর রিয়েল ওয়ার্ল্ড প্রোগ্রাম নেই। কারণ ব্যবহারকারীর অন্যান্য উদ্দেশ্য পূরণের জন্য প্রোগ্রামের আনুগ পরিবর্তন হতে পারে কোন বিকল্প থাকে না। আর এভাবেই প্রোগ্রামারদের সমস্ত শ্রম পুণশ্রম হয়ে যায়। আর এ জন্যই এক প্রতিভাধর সফটওয়্যার উদ্ভাবক বলেছেন, সফটওয়্যার দুবার লেখা উচিত, একবার সমস্যাটি বোঝার জন্য এবং আর একবার সমস্যার সমাধানের জন্য।

একটি কঠমাইজ অ্যাপ্লিকেশন যেখানে যতই ভাল হোক না কেন ব্যবহারকারীর সমোচন পরিবর্তন, পরিমার্জন, পরিবর্তন চাইবেনই। এতে কোন সন্দেহ নেই এবং এটাই স্বাভাবিক। আর পরিবর্তন, পরিমার্জন বা পরিবর্তন শুখু এক বা একাধিক মডুল বা কম্পোনেন্ট এর সংযোজন নয়। বরং সম্পূর্ণ পরিমার্জন। হয়তোবা দেখা যাবে ডাটাবেইজ ট্রাকচার এর পরিবর্তন হবে। আর এ পরিবর্তনের সংগে সংগে আসবে এর স্কীপ-এর পরিবর্তন, validation routine-এর পরিবর্তন এবং

এরকম আবার অনেক কিছু। আর এ পরিবর্তন শুরু হবে সোর্স কোড-এর শুরু থেকে শেষ পর্যন্ত। আর এই পরিবর্তনের ফলে পূর্বের সলিড সোর্স এ মেমো থেকে নতুন নতুন bugs বা ভুল। আর এ কারণে পরিবর্তন হলো অতীব বিপদজনক।

ষ্ট্রাকচার প্রোগ্রামিং আর একটা সমস্যা হলো যে একে সরাসরি পুনঃব্যবহার করা যায় না। কারণ এ পদ্ধতিতে প্রোগ্রামাররা একটা নির্দিষ্ট প্রকল্পে নিয়ে কাজ করেন এবং প্রকল্পেই সাপোর্টেই বিভিন্ন মডুল বা কম্পোনেন্ট পরিস্ফুটন করেন। ফলে একটা জেনারেল বা universal reusable মডুল বা কম্পোনেন্ট তৈরী হয় না। এবং স্বাভাবিকভাবে মডুলটিকে অন্যান্য এপ্লিকেশনে পরিবর্তন বা পরিমার্জন ব্যতীত ব্যবহার করা যায় না। তাছাড়া পরিবর্তনে থাকে অনেক অনেক bug বা ভুলের ঝুঁকি, যা পূর্বেই উল্লেখিত হয়েছে।

অন্যদিকে, একটা সফটওয়্যার হাউজে একজন সফটওয়্যার এণ্ড সিস্টেম ম্যানেজার মডুল বা কম্পোনেন্ট পুনঃব্যবহারের ঝুঁকি নিয়ে থাকেন। এমন অবস্থায় সফটওয়্যার এণ্ড সিস্টেম ম্যানেজার তাঁর অধীন, production programmers generalised code লেখায় উৎসাহিত করেন। এবং জেনারেলাইজ কোড লেখার জন্য সফটওয়্যার এণ্ড সিস্টেম ম্যানেজারকে বেশ পছন্দ করতে হয়। কারণ কম্পোনেন্ট বা মডুল ডিভাইস উদ্ভাবনে প্রয়োগক্ষেত্র, ব্যবহারেরই নানাবিধ বিধেয় প্রয়োজন প্রোগ্রামারদেরকে প্রশিক্ষণ, নির্দেশ

এবং তত্ত্বাবধান ইত্যাদির দায়িত্ব সফটওয়্যার এণ্ড সিস্টেম ম্যানেজারকে নিতে হয়। তাছাড়া জেনারেলাইজ কোড লিখলে অপেক্ষাকৃত অনেক অনেক বেশী সময়ের প্রয়োজন হয়। আর এ সময় বিনিয়োগ লাভজনক কিনা, এটাও একজন সফটওয়্যার এণ্ড সিস্টেম ম্যানেজার হাজার বার ভাবে চিনে। আর জেনারেলাইজ কোড লিখলে শুধুমাত্র সফটওয়্যার হাউজের পক্ষেই সম্ভব যদি সময় বিনিয়োগ লাভজনক হয়। এছাড়া একক প্রোগ্রামাররা জেনারেলাইজড রিইউজবল কোড নিয়ে আদৌ খুব একটা ভাবেন না। ফলে সম্ভবিত ভাবে একটা দেশে প্রচুর সময়ের অপচয় হয়। আর এ জন্যই আমাদের প্রয়োজন সুস্থ সুবিন্যস্ত প্রশিক্ষণ পদ্ধতি এবং এর নীতিমালা। নতুবা সফটওয়্যার পরিস্ফুটনে ব্যয়িত সময় ও শ্রমের বহুলাংশই পশুশ্রমে পরিণত হবে। এ বিষয়ে আলোকপাত এ আলোচনার বিষয়বস্তু নয় বিধায় আমরা মূল প্রশ্নে ফিরে যাব এবং সময় ও সুযোগ হলে এ বিষয়ে ভিন্নভাবে বিস্তারিত আলোচনা করব।

প্রোগ্রামিং-এর বিবর্তনের দিকে একটু দৃষ্টি দিলেই আমরা দেখতে পাই যে এর বিবর্তন ধারায় এক বিরাট অংশে জুড়ে আছে প্রোগ্রাম লাইব্রেরী তৈরী ও প্রয়োগ। আর এর মূল লক্ষ্য ও উদ্দেশ্য হলো পুনঃব্যবহার বা Reusability.

অহলে প্রোগ্রাম লাইব্রেরী কি? একটা লাইব্রেরী হলো অনেকগুলো জেনারেল রুটিন বা মডুল অথবা ফাংশন এর সমষ্টি। আর এদের প্রতিটি

ফাংশনের রয়েছে সঠিকভাবে উল্লেখিত উদ্দেশ্য এবং ব্যবহারের নিয়মাবলী। এ ধরনের লাইব্রেরীর রয়েছে বিপুল চাহিদা এবং বিশ্বব্যাপী বাজার। এবে যে কোন প্রোগ্রামিং ল্যাংগুয়েজ এর জন্য এটা প্রয়োজ্য। আপনি যে প্রোগ্রাম ল্যাংগুয়েজের কমপাইলার ব্যবহার করুন না কেন, আপনি এমন শত শত লাইব্রেরীর খোঁজ পাবেন, সেখানে আপনি পাবেন গাণিতিক বিশ্লেষণ, পরিসংখ্যান সম্পর্কীয় ফাংশন মেনু তৈরীকরণ, file manipulation এমন কি কোমিউনিকেশন ইন্টারফেস এবং tools। এ ধরনের লাইব্রেরী সফটওয়্যার পরিস্ফুটনে বাঁচিয়ে দেয় সময় শ্রম এবং টাকা পয়সা। আর এ সুবিধা জোগ করছে বিশ্বব্যাপী প্রচুর প্রতিষ্ঠান।

XBase dialect-এ Clipper 5.01-র Add-on লাইব্রেরীর রয়েছে বিপুল চাহিদা। প্রোগ্রাম ল্যাংগুয়েজ হিসাবে ট্রিপার ইতিমধ্যেই অর্জন করেছে এক বিশ্বব্যাপী জনপ্রিয়তা। আমরা মনে করি ট্রিপার-এর জনপ্রিয়তা উৎসাহিতকারী বৃত্তি পাবে এর শিক্ষনের সময় ও উৎসাহনের সুবিধাজনক অবস্থানের জন্য। আর এ জন্যই বাংলাদেশের প্রোগ্রামারদের জন্য আমাদের এ প্রয়াস। আগামী সংখ্যায় আমরা আলোকপাত করব অর্থাৎ অরিয়েন্টেড প্রোগ্রামিং-এর বিবিধ সুবিধাবলী এবং OOP হিসাবে ট্রিপার-এর অবস্থান ইত্যাদি।

— চলেবে

## TOTAL SERVICES

Exporter, Importer, Manufacturer, Commission Agent, Supplier, Rent A Car, Computer Training, DTP, Photocopier, Spairal, Short-Film, Garments Accessories, Printing & Stationary, TV Antenna, Switch, Socket, Toys, Pipe, Trolley, Plastic Cover, Bottles & Handicrafts etc.

Sales	Rent & Services	Data Entry
-------	-----------------	------------

Computer Printer	Computer Printer	Bio-data
Stabilizer	H/W Install	Thesis/Letter
UPS/Fax	Consultancy	Payroll/GL
Diskette	Software Dev.	Reports & DTP
Ribbon	Ribbon Re-inking	Stock/LC
Paper	Ribbon Re-filling	Field Report
		Statistical data

## TRAINING

WordPerfect 5.1	Telex	Basic Programming
Word Star	Fax	dBase Programming
Lotus 1-2-3	Typing	Turbo-C
Quattro Pro 3.0	Driving	Pascal/Cobol
dBase III Plus / IV	Forshand	Fortran-77
Accounting	Sewing	Spss pc+



Top Of The Time

## ANANTA JOTI

Baltush Sharaf Mosque  
Farmgate (Opposite Trigon PS)  
149/A, AirPort Road, Dhaka-1215.

Phone: 815445, 814253  
Fax : 810-82-814253

## UPS ! UPS !!

WHEN YOU BUY AN UPS, PLEASE REMEMBER THE FOLLOWING POINTS :

- its input regulation
- output regulation
- power factor and
- backup time

We are offering you, ranges of UPS from Taiwan & HongKong - Call Power, Prostar, Centralion, US Power, etc. World famous products.

## DETO SEARCH

MIRPUR 10-B, AVE.1 / PLOT 3  
DHAKA-1221  
PHONE : 802458, 802763

Your Trusted Computer Dealer Since 1982