

IDL

Interactive Data Language

Curso do IDL

- * Coordenador: Lucio Marassi de Souza Almeida
- * Professor: Antonio Paulo V. Pinto
- * Monitores:
 - * Sérgio Filipe Gadelha Roza
 - * Ewerton César Barros Filgueira

Curso do IDL

- * Local:
 - * UFRN (Universidade Federal do Rio Grande do Norte).
 - * ECT (Escola de Ciências e Tecnologia), 1º andar, sala 7.
- * Horários: 46M56
 - * Quartas e Sextas das 10:50 às 12:30
- * Duração:
 - * Aproximadamente 21 aulas, 42 horas.

Aulas

- * Expositivas
- * Análises de código-fonte
- * Atividades
- * Ritmo do aluno/Aula para o aluno

Curso do IDL

- * Blog:

- * <http://idltutorial.blogspot.com>

- * E-mail:

- * antoniopaulovp@gmail.com

- * Download do IDL:

- * <http://docs.astro.columbia.edu/files/idl/7.0/>

Curso do IDL

- * 11 semanas de aula, totalizando 21 aulas.
- * Dividido em 4 Etapas:
 - * Introdução à programação
 - * Introdução ao IDL
 - * Programação Estruturada em IDL
 - * Programação Vetorizada em IDL

Introdução à Programação

- * Conceitos fundamentais para programação.
- * Desenvolvimento de raciocínio para solução de problemas.
- * Revisão de álgebra booleana.
- * Revisão de álgebra matricial.

Introdução ao IDL

- * Introdução ao ambiente do IDL
- * Sintaxe básica
- * Variáveis
- * Compilação mental

Programação estruturada

- * Programação condicional no IDL
- * Programação de Loops em IDL
- * Atividades de treinamento em programação
- * Estruturação de projetos

Programação vetorial

- * Vetorização de condicionamentos
- * Vetorização de loops
- * Transposição estruturada/vetorial
- * Otimização e busca de falhas em código

Desenvolvedores

- * IDL – Interactive Data Language
- * ITT Exelis
 - * <http://www.exelisinc.com>
- * ITT Visual Information Solutions, Inc.
 - * <http://www.itt.com>

Aplicações

- * Matemática
- * Física
- * Estatística
- * Engenharia
- * Simulações e tratamento de dados em geral...

Quem usa o IDL?

- * National Center for Atmospheric Research
- * University of Michigan
- * University of Colorado
- * Rutherford Appleton Laboratory

Quem usa o IDL?

- * NASA's Goddard Space Flight Center
- * Ball Aerospace & Technologies Corp
- * European Space Agency
- * Naval Research Laboratory
- * Entre diversas outras...

Vantagens

- * Portabilidade:
 - * Windows
 - * Linux
 - * UNIX/Solaris
 - * Mac OS
- * Banco de Dados
- * Conectividade TCP/IP

Vantagens

- * Paralelismo local natural
- * Possibilidade de paralelismo em rede
- * Sem limitação de termos em vetores
 - * Obs: Linguagens de alto níveis não aceitam mais do que $2^{16} \cong 65k$ termos, como o MatLab.

Repositórios

- * NASA

- * <http://idlastro.gsfc.nasa.gov/>

- * http://hesperia.gsfc.nasa.gov/hessi/solar_cd/FAQ/IDL_FAQ.htm

- * Center for Solar-Terrestrial Research (USA)

- * <http://solar.njit.edu/~cdenker/idl/>

Repositórios

- * National Centre for Atmospheric Science (UK)
 - * http://cms.ncas.ac.uk/cms_site_old/component/option,com_dbquery/Itemid,245/task,ExecuteQuery/qid,25/previousTask,PrepareQuery/
- * Coyote's Guide to IDL Programming
 - * <http://www.idlcoyote.com/>

Velocidade

- * Linguagens de Alto nível:
 - * Fáceis e rápidas de programar, lentas de se executar.
 - * Exemplos: MatLab, Mathematica, Mapple, ...
- * Linguagens de Médio/Alto:
 - * Difíceis e demoradas de programar, rápidas de executar.
 - * Exemplos: C/C++, Fortran, ...

Velocidade

- * O IDL possui aspectos de ambos os níveis.
 - * Tão fácil quanto as de alto nível
 - * Obs: Se usado em níveis avançados torna-se ainda mais fácil que as de alto nível.
- * Quase tão rápida quanto as de médio nível.
 - * Isso torna-se uma grande vantagem em relação as de alto nível, e sua simplicidade torna-a vantajosa em relação as de médio nível.

Velocidade é tão importante assim?

- * Simulações para atividades:
 - * Costumam demorar alguns minutos.
 - * Lista de atividades simples de graduação.
- * Simulações reais:
 - * Demoram horas, dias, meses, anos.
 - * Pesquisas, mestrados, doutorados, empresas.

O quão rápido o IDL é?

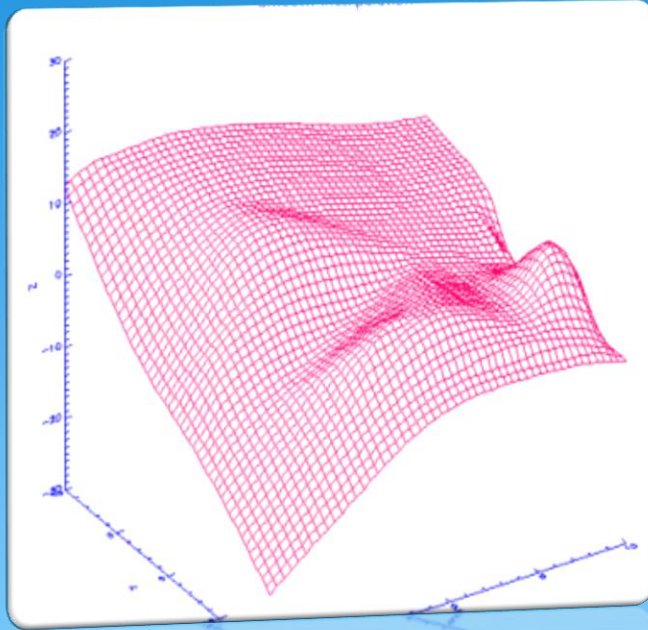
- * O IDL possui diversas formas de se programar.
 - * Estruturada
 - * Mais comum.
 - * Mostra a execução passo a passo.
 - * Códigos geralmente grandes
 - * Diversos minutos/horas para se programar.
 - * Vetorizada
 - * Mais veloz.
 - * Mais simples/intuitivo.
 - * Poucas linhas de código fonte.
 - * Poucos segundos/minutos para se programar.

IDL estruturado versus Mathematica

- * Mesma simulação implementada em ambos os ambientes.
- * IDL estruturado:
 - * Simulação concluída.
 - * 20 minutos
- * Mathematica:
 - * Simulação teve de ser interrompida.
 - * 60 minutos.
 - * 20% Concluída.

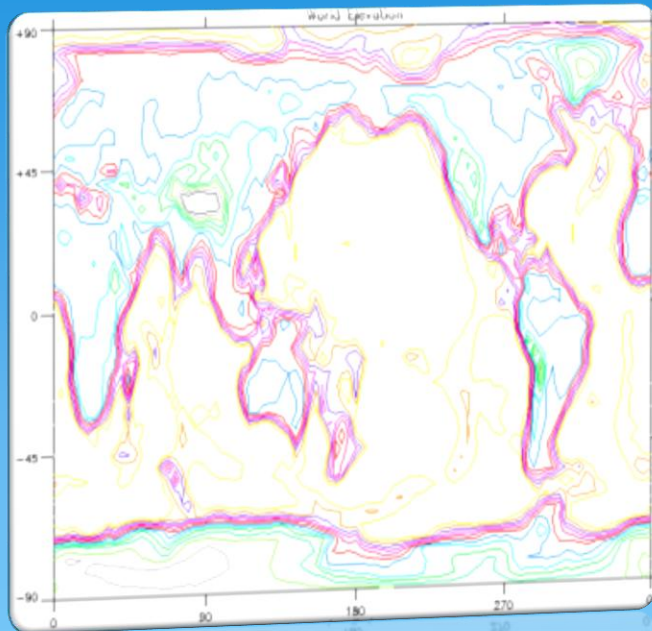
IDL: Estruturado versus Semi-Vetorizado (75% vetorizado)

- * Mesma simulação.
- * IDL Estruturado:
 - * Concluída em mais de 13 horas.
- * IDL Semi-Vetorizado
 - * Concluída em aproximadamente 40 minutos.



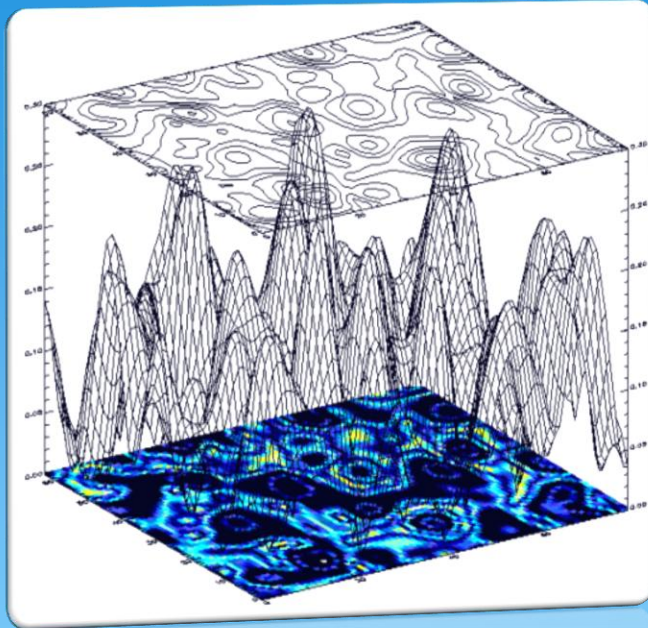
Gerador de gráficos

Assim como muitas das linguagens de alto nível atuais o IDL possui um gerador de gráficos próprio.



Vasta gama de gráficos

O IDL possui uma variedade enorme de tipos de gráficos para se utilizar, incluindo além dos simples gráficos de retas, também gráficos de contornos.



Mesclagem de gráficos

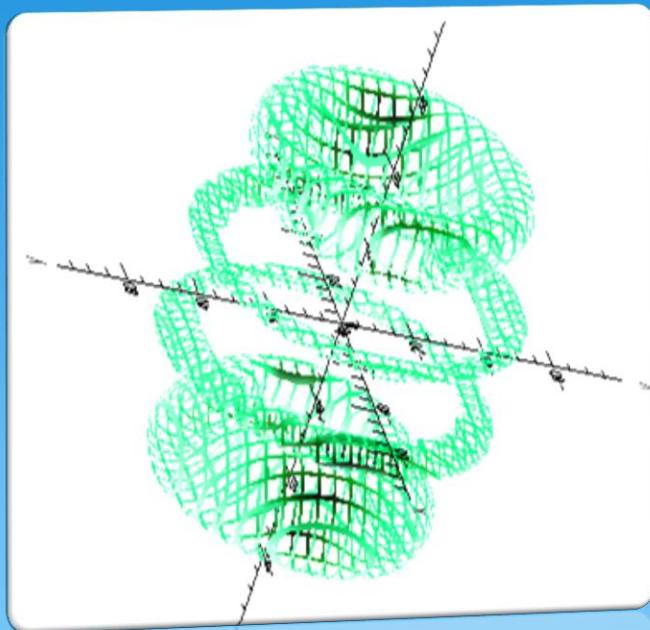
Alguns gráficos possibilitam serem utilizados em conjunto com outros afim de mostrarem informações mais completas e perceptíveis.



Gráficos dinâmicos

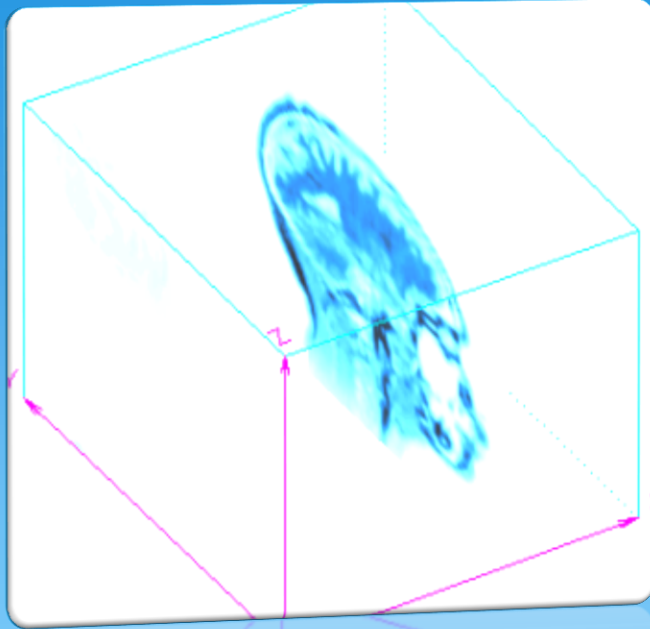
Ainda há também tipos de gráficos que são dinâmicos, ou seja, o usuário pode interagir com o gráfico.

Ao lado vemos uma superfície onde o usuário poderá escolher a posição de cada curva de contorno, e quantas curvas deseja exibir.



Gráficos de volumes

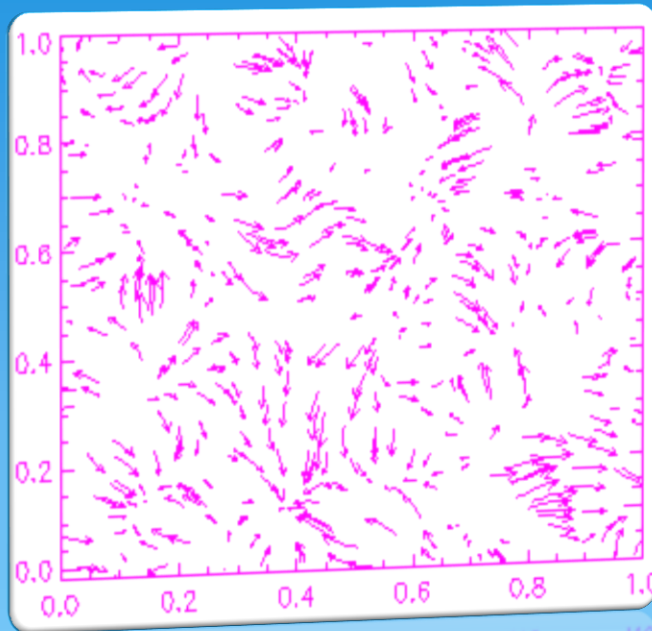
É possível criar também volumes tridimensionais para simular determinadas objetos.



Gráficos de densidade e composição de sólidos

Este tipos avançado de gráfico possibilita fazer cortes em determinado sólido para verificar a sua composição e/ou densidade vista daquele corte.

O cubo ao lado permite fazer um corte em qualquer direção e ver a partir desta fatia o interior do crânio humano.



Campos vetoriais

Com diversos usos e aplicações, este é um tipo de gráfico de grande importância e que o IDL disponibiliza com diversas opções de configurações.



Simulações dinâmicas

Gráficos tridimensionais que se movem para mostrar o progresso da simulação.

A imagem ao lado é de uma simulação de órbita de um satélite, no gráfico real pode-se ver o satélite girando em torno da terra.



Gráficos dinâmicos com interação do usuário

O gráfico ao lado é um simulador de voo, onde o usuário pode girar a câmera e mover-se semelhante ao que faria caso estivesse em um avião.

Há também gráficos panorâmicos, onde o usuário escolher de que ponto do mapa deseja olhar e em que direção.

Cronograma de aulas

1ª Semana

- * Aula 00 – Dia 21 de Março de 2012
 - * Apresentação do Curso, apresentação do IDL e apresentação do cronograma de aulas.
- * Aula 01 – Dia 23 de Março de 2012
 - * Algoritmo, conceitos de programação, revisão de álgebra booleana e revisão de álgebra matricial.

2ª Semana

- * Aula 02 – Dia 28 de Março de 2012
 - * Pensamento, organização e solução de problemas através da elaboração de fluxogramas.
- * Aula 03 – Dia 30 de Março de 2012
 - * Ambiente de trabalho, conceitos básicos, variáveis, operações matemáticas, comparadores, operações booleanas, maior/menor de dois e operadores especiais.

3ª Semana

- * Aula 04 – Dia 04 de Abril de 2012
 - * Criando o primeiro procedimento e operadores incrementais.
- * Aula ?? – Dia 06 de Abril de 2012
 - * Aula cancelada devido ao feriado da semana santa. 5 e 6 de Abril.

4ª Semana

- * Aula 05 – Dia 11 de Abril de 2012
 - * Arrays, vetores, matrizes, operações matriciais e plotagem.
- * Aula 06 – Dia 13 de Abril de 2012
 - * Bases da programação estruturada, estruturas de condicionamento, IF...THEN...ELSE.

5ª Semana

- * Aula 07 – Dia 18 de Abril de 2012
 - * Cadeia IF...ELSE...IF, condicionamento embutido ?: e estruturas de seleção switch/case.
- * Aula 08 – Dia 20 de Abril de 2012
 - * Estruturas de loops, GOTO e FOR.

6ª Semana

- * Aula 09 – Dia 25 de Abril de 2012
 - * Estrutura FOR e WHILE.
- * Aula 10 – Dia 27 de Abril de 2012
 - * Estrutura REPEAT...UNTIL e comparação de estruturas de loops.

7ª Semana

- * Aula 11 – Dia 02 de Maio de 2012
 - * Procedimentos, entradas/saídas, funções e scripts.
- * Aula 12 – Dia 04 de Maio de 2012
 - * Procedimentos, entradas/saídas, funções e scripts.

8ª Semana

- * Aula 13 – Dia 09 de Maio de 2012
 - * Variáveis globais/locais, compartilhamento de variáveis e a estrutura COMMON.
- * Aula 14 – Dia 11 de Maio de 2012
 - * Estrutura COMMON e a leitura/escrita de arquivos.

9ª Semana

- * Aula 15 – Dia 16 de Maio de 2012
 - * Aplicações com COMMON e leitura/escrita de arquivos.
- * Aula 16 – Dia 18 de Maio de 2012
 - * Introdução a vetorização, o que é, para que serve, como funciona, limitações e vantagens.

10ª Semana

- * Aula 17 – Dia 23 de Maio de 2012
 - * Vetorizando estruturas de condicionamentos.
- * Aula 18 – Dia 25 de Maio de 2012
 - * Vetorizando estruturas de loops.

11ª Semana

- * Aula 19 – Dia 30 de Maio de 2012
 - * Transposição loops x vetores
- * Aula 20 – Dia 1º de Junho de 2012
 - * Depuração de erros simples, planejamento/projeto de simulações e otimização.

Bibliografia

- * Wikipédia

- * <http://en.wikipedia.org>

Dúvidas?

E-mail: antoniopaulovp@gmail.com

Blog: <http://idltutorial.blogspot.com>