

PROJETO 2

Este projeto estende o projeto anterior. São acrescentados novos comandos e novas consultas.

A colonização espacial já está em estágio avançado. O ser humano habita diversos planetas da Via Láctea. Recentemente, Elon Musk XVII inaugurou a primeira colônia, batizada de New Grimesland (NG), em um aprazível planeta em Andrômeda. Infelizmente, descobriu-se que o planeta é ocasionalmente atingido por meteoros que não são completamente desintegrados na entrada da atmosfera. Para agravar a situação, os meteoritos residuais são muito radioativos. Cada meteorito produz de 1 a 25mSv no momento do impacto.

A colônia possui várias edificações. Uma parede reduz em 20% a radiação. Por isso, quando uma chuva de meteoros é detectada antecipadamente, um alerta é emitido e os habitantes tentam se abrigar na edificação mais próxima.

Sua tarefa é identificar, para cada indivíduo, a edificação mais próxima e o eventual nível de exposição à radiação. A tabela abaixo mostra o efeito de diferentes níveis de exposição. Ela determina as classes de exposição.

Exceder a taxa de doses de radiação e efeitos

(adaptado de: <http://pt.nextews.com/8893167c/>)

Dose única (mSv)	O que acontece com o corpo
Até 25 (00ffff)	Alterações no estado de saúde não são observadas
25-50 (00ff00)	Reduziu o número total de linfócitos (imunidade reduzida)
50-100 (ff00ff)	Uma redução significativa nos linfócitos, sinais de fraqueza, náuseas, vômitos
100-250 (0000ff)	Em 5% dos casos ser fatal, mais observado chamado ressaca radiação (sintomas semelhantes a ressaca álcool)
250-500 (800080)	Alterações no sangue, esterilização masculina temporária, 50% de mortalidade dentro de 30 dias após a exposição
500-1000 (000080)	Uma dose letal de radiação, não de ser tratada
1000-8000 (ff0000)	Coma e morte dentro de 5-30 minutos
mais de 8000 (000000)	Morte instantânea a partir de um feixe

ENTRADA DE DADOS

A entrada de dados, via de regra, ocorrerá por meio de um ou mais arquivos. Estes arquivos estarão sob um diretório, referenciado por **BED** neste texto.¹

SAIDA DE DADOS

Os dados produzidos serão mostrados na saída padrão e/ou em diversos arquivos-texto. Alguns resultados serão gráficos no formato SVG. Os arquivos de saída serão colocados sob um diretório, referenciado por **BSD** neste texto.²

DESCRIÇÃO

A entrada do algoritmo será basicamente um conjunto de retângulos e círculos dispostos numa região do plano cartesiano e algumas consultas, por exemplo, que indagam se dois retângulos se sobrepõem. Os comandos estão contidos num arquivo **.geo** e as consultas num arquivo **.qry**.

Considere a Figura 1. Cada retângulo é definido por uma coordenada âncora (veja ponto roxo na figura) e por suas dimensões. A coordenada âncora do retângulo é seu canto inferior esquerdo³ e suas dimensões são sua largura (w) e sua altura (h). Cada retângulo é identificado por um código alfa-numérico.

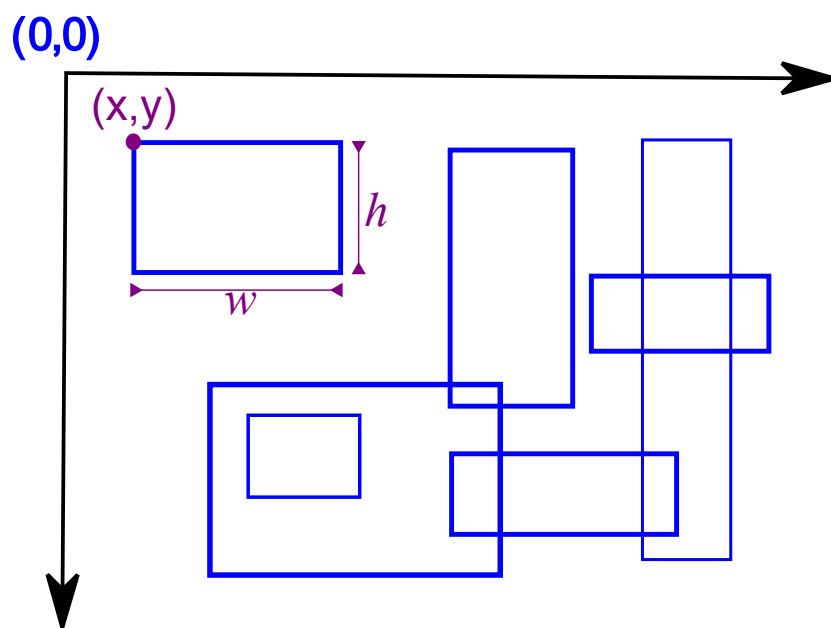


Figura 1: Retângulos no plano

As tabelas abaixo mostram os formatos dos arquivos de entrada (.geo e .qry). Os arquivos de entrada são compostos, basicamente, por conjunto de comandos (um por linha).

1 Indicado pela opção -e.

2 Indicado pela opção -o.

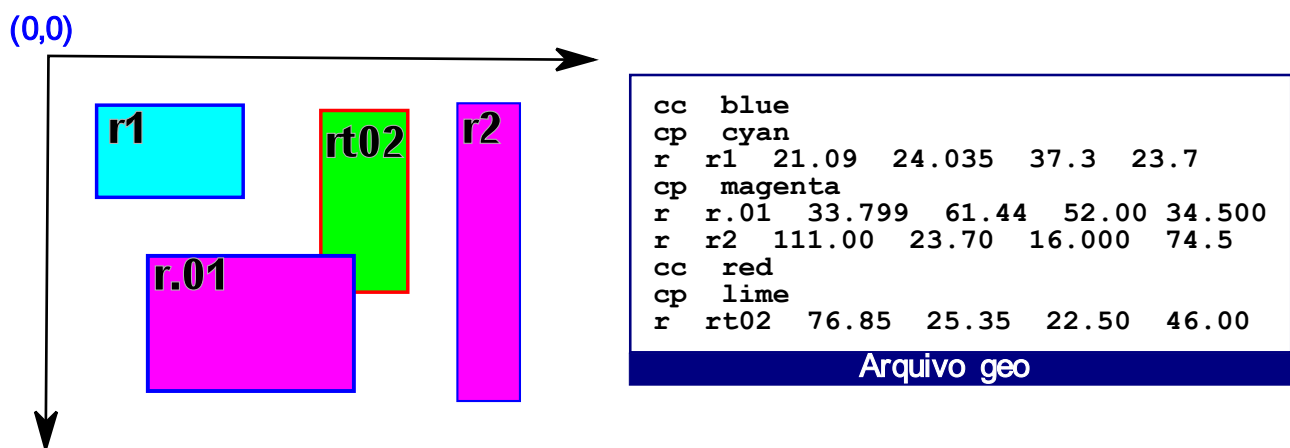
3 Note que o plano cartesiano está desenhado "de ponta-cabeça" em relação à representação usual.

Cada comando tem um certo número de parâmetros, separados por um espaço. Os parâmetros mais comuns são:

- **id**: um identificador alfa-número. Por exemplo, r18, retang-0.1.2.3, etc.
- **w, h**: números reais. Dimensões do retângulo.
- **x, y**: números reais. Coordenada (x,y).
- **cor**: string. Cor válida dentro do padrão SVG.⁴ Em lugar da cor, pode ser colocado a caractere @, indicando que não deve ser usada nenhuma cor.

comando	parâmetros	descrição
nx	n	<i>Número aproximado de retângulos</i>
cc	cor	<i>Cor para o contorno dos retângulo</i>
cp	cor	<i>Cor preenchimento dos retângulos</i>
bc	cor	<i>Cor do contorno dos círculos</i>
pc	cor	<i>Cor do preenchimento dos círculos</i>
r	id x y w h	<i>Desenhar retângulo: w é a largura do retângulo e h, a altura</i>
c	id x y r	<i>Desenhar círculo com centro em (x,y) e raio r</i>
comandos .geo		

Abaixo, um exemplo de um arquivo geo, com sua respectiva representação pictórica. Vamos supor que o nome do arquivo seja a1.geo.



As consultas estão descritas na tabela abaixo. A primeira coluna da tabela apresenta códigos que indicam o efeito esperado da consulta:

- **D**: (deleção): Dados são removidos de estruturas de dados.

⁴ <http://www.december.com/html/spec/colorsvg.html>.
<https://www.w3.org/Graphics/SVG/IG/resources/svgprimer.html>

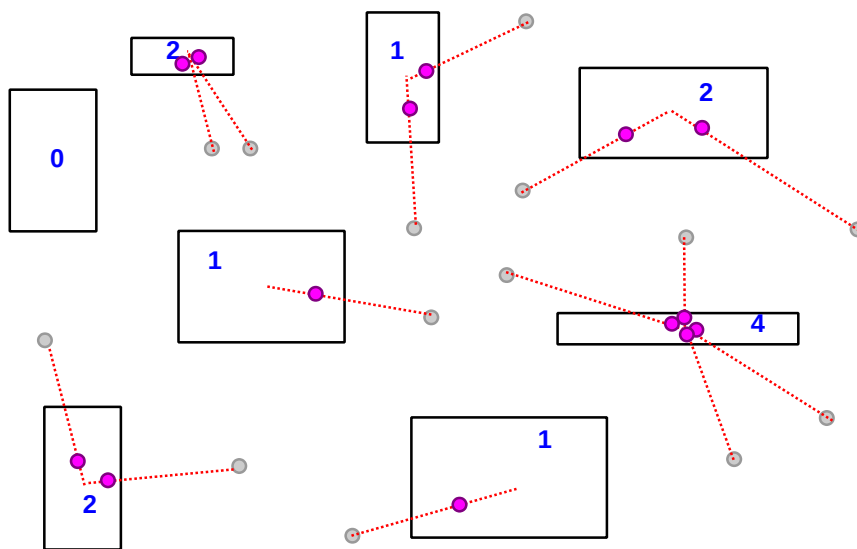
- U (update): Dados existentes são modificados
- I (inserção): Novo dado é inserido
- C (consulta): estrutura de dados é consultada, mas não modificada
- E (exposição): algum elemento gráfico é colocado na saída svg, mas não é armazenado .

Note que muitas consultas são semelhantes. Espera-se que as partes comuns sejam fatoradas e reaproveitadas.

	comando	parâmetros	descrição
D	dpi	x y	Remover todos os retângulos para os quais o ponto (x,y) é interno. TXT: reportar os identificadores dos retângulos removidos. SVG: retângulo removido não deve aparecer
D	dr	id	Semelhante ao dpi, mas remove retângulos que estejam inteiramente dentro do retângulo de identificador id. Não remove o triângulo id.
U E	fg	x y r	Pessoas que estão dentro da região delimitada pela circunferência com centro em (x,y) e raio r abrigam-se na edificação mais próxima. Calcular a distância do centro de massa do retângulo ao centro do círculo. TXT: para cada edificação (colocar id do retângulo), listar as pessoas abrigadas (os identificadores dos círculos) SGV: colocar círculo em cinza claro na posição original, colocar linha tracejada da posição original até dentro da edificação (retângulo) mais próximo, mover (animação) o círculo da posição inicial até a posição final. Escrever o número de abrigados na edificação
I U E D	im	x y s	Meteoro com radiação de s mSv impactou o ponto (x,y). Calcular classe de exposição à radiação e preencher respectivo círculo com a cor da tabela. As pessoas com morte instantânea devem ser removidas (representadas por um círculo preto e com cruz branca). A exposição é cumulativa. SVG: colocar círculo cinza escuro na coordenada (x,y). O raio deve ser proporcional à radiação s. Pintar círculo referente à pessoa conforme nível de exposição. Assinalar mortes instantâneas como descrito acima. TXT: listar pessoas (id do círculo) das pessoas com morte instantânea e morte iminente.
D	t30		Transcorreram 30 minutos. As pessoas da classe de morte iminente morreram. SVG e TXT: semelhante à morte instantânea.

C E	nve	x y	Nível de exposição corrente no ponto (x,y). SGV: colocar um pequeno quadrado com bordas arredondadas, contendo o nível de exposição corrente da região onde está o ponto e pintá-lo com a cor correspondente à classe de exposição. TXT: reportar o nível
	comandos de consulta (.qry)		

Exemplo da consulta fg



IMPLEMENTAÇÃO

Uma **árvore k-d** (abreviação para a árvore k-dimensional) é uma estrutura de dados de particionamento do espaço para a organização de pontos em um k-dimensional espaço. Árvores k-d são estruturas úteis para uma série de aplicações, tais como pesquisas envolvendo pesquisa multidimensional de chaves (e.g. busca de abrangência e busca do vizinho mais próximo). Árvores k-d são um caso especial de árvores de particionamento binário de espaço.

Uma árvore k-d é uma árvore binária em que cada nó é um ponto k-dimensional. Cada nó não-folha pode ser considerado implicitamente como um gerador de um hiperplano que divide o espaço em duas partes, conhecido como semiespaço. Os pontos à esquerda do hiperplano são representados pela subárvore esquerda desse nó e pontos à direita do hiperplano são representados pela subárvore direita. A direção do hiperplano é escolhida da seguinte maneira: cada nó na árvore é associado a uma das k-dimensões, com o hiperplano perpendicular a esse eixo dimensional. Assim, por exemplo, se para uma determinada operação de split o eixo "x" é escolhido, todos os

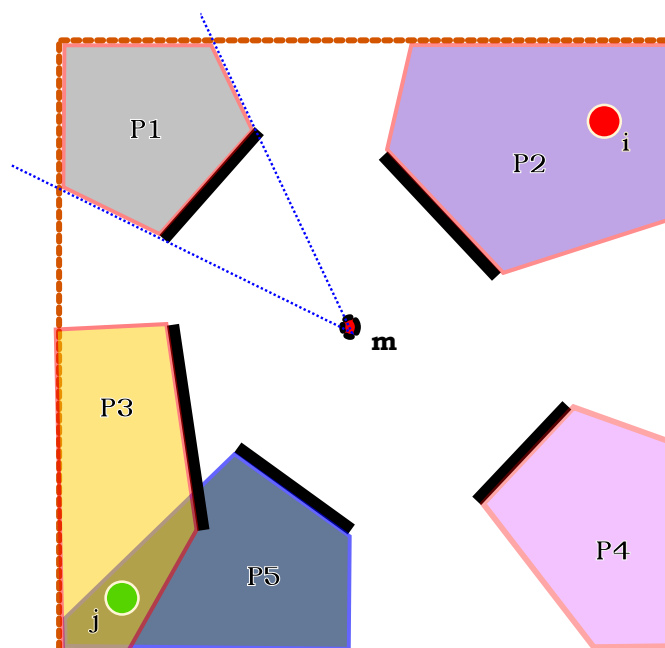
pontos da subárvore com um valor " x " menor que o nó irão aparecer na subárvore esquerda e todos os pontos com um valor " x " maior vão estar na subárvore direita. Nesse caso, o hiperplano seria definido pelo valor de x do ponto, e o seu normal seria a unidade do eixo x .

Copiado da Wikipedia: https://pt.wikipedia.org/wiki/%C3%81rvore_k-d

- Armazenar retângulos, círculos em árvores-Kd
- Armazenar segmentos ativos em árvores binária busca
- Armazenar polígonos em um dos dois tipos de árvores
 - Dica: usar mínimo retângulo envolvente
- Usar o algoritmo **qsort** da biblioteca do C
- Dica: Para determinar o nível de exposição de um ponto, verificar a quantos polígonos ele é interno.
- Usar o algoritmo de cálculo de região de visibilidade para determinar a área de máxima exposição
- Usar a implementação dinâmica duplamente encadeada da lista.
- Quando fizer busca em uma região, só descer em sub-árvores “promissoras”.
- É **expressamente proibido** declarar structs em arquivos .h.
- Coloque um comentário nos arquivos txt e svg informando o nome do aluno.

Atenuação da radiação (Sugestão):⁵

A figura abaixo mostra um meteorito caído. Note que os segmentos foram “cercados” por um retângulo (em laranja). Se traçarmos semirretas do meteorito passando pelas extremidades de um segmento e determinamos as intersecções com o retângulo envolvente, obtemos um polígono convexo que corresponde à região de atenuação daquela parede (veja, por exemplo, o polígono P1).



⁵ Mateus, obrigado pela inspiração.

A sugestão é, para cada meteorito, criar uma árvore dos polígonos relativos às “sombras” provocadas por cada parede. Dado um ponto, para determinar a atenuação total basta saber a quantos destes polígonos ele é interno. Por exemplo, o ponto **i** é interno apenas ao polígono P2, portanto a radiação do meteorito **m** é atenuada por uma parede (-20%). Já o ponto **j** é interno a dois polígonos (P3 e P5) e, assim, sofre duas atenuações.

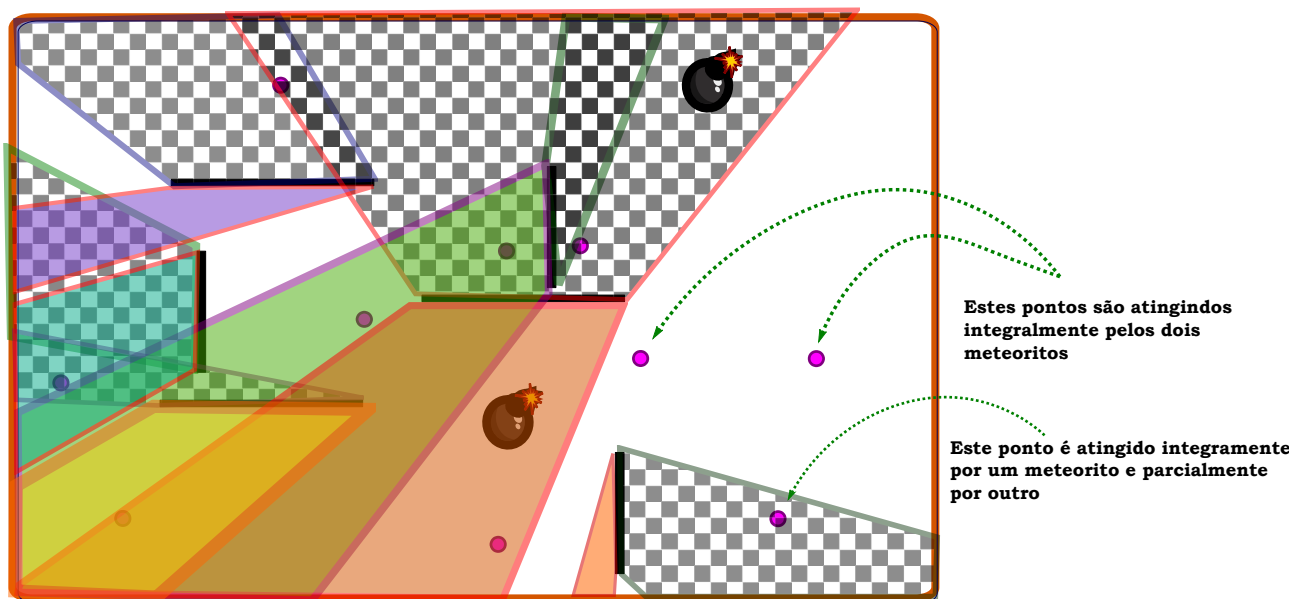
Note que o polígono que determina a região de visibilidade a partir do ponto **m** possui incidência máxima de radiação.

Cálculo da Incidência Total

A incidência total de radiação em um ponto é a soma simples da incidência de cada meteorito. Para cada meteorito é necessário determinar se ele está na região de visibilidade do meteorito (incidência máxima) ou se ele está na sombra do meteorito (cálculo da atenuação).

Assim, uma sugestão poderia ser:

- como já mencionado, manter uma árvore dos polígonos de sombras para cada meteorito.
 - ponto interno a polígono: atenuação (combinação de atenuações diminui radiação)
- manter outra árvore com os polígonos das regiões de visibilidade
 - ponto interno à polígono: intensificação. Somar a radiação de cada polígono de visibilidade para o qual o ponto é interno
- Somar a contribuição de cada árvore



ORGANIZAÇÃO DA ENTREGA

O trabalho deve ser submetido no formato **ZIP**, cujo nome deve ser curto, mas suficiente para identificar o aluno ou a equipe.⁶ Este arquivo deve estar organizado como descrito à frente.

PROCESSO DE COMPILAÇÃO E TESTES DO TRABALHO

Organização do ZIP a ser entregue

A organização do zip a ser entregue pelo aluno deve ser a seguinte:

[abreviatura-nome]

LEIA-ME.txt

Por exemplo, josers.

colocar matrícula e o nome do aluno. Atenção: O número da matrícula de estar no início da primeira linha do arquivo. Só colocar os números; não colocar qualquer pontuação.

*

Outros arquivos podem ser solicitados.

/src

(arquivos-fonte)

makefile

*deve ter target para a geração do arquivo objeto de cada módulo e o target **progr** que produzirá o executável de mesmo nome dentro do mesmo diretório **src**. Os fontes devem ser compilados com a opção **-fstack-protector-all**.*

** adotamos o padrão C99. Usar a opção **-std=c99**.*

*.h e *.c

***Atenção:** não devem existir outros arquivos além dos arquivos fontes e do makefile*

Organização do diretório para a compilação e correção dos trabalhos (no computador do professor):

[HOME DIR]

*.py

scripts para compilar e executar

\t

diretório contendo os arquivos de testes

*.geo *.qry

*arquivos de consultas, em geral, distribuídos em alguns outros subdiretórios. Por exemplo, as consultas relativas a um arquivo **al.geo**, normalmente, estarão no subdiretório **al**.*

\alunos

(contém um diretório para cada aluno)

\abrnome

*diretório pela expansão do arquivo submetido (p.e., **josers**)*

*outros subdiretórios para os arquivos de saída informados na opção **-o***

Os passos para correção serão os seguintes:

⁶ Por exemplo, josers.zip (se aluno se chamar José Roberto da Silva), josers-mariabc.zip (para uma equipe com dois alunos. Evite usar maiúsculas, caracteres acentuados ou especiais.

1. O arquivo .zip será descomprimido dentro do diretório alunos, conforme mostrado acima
2. O makefile provido pelo aluno será usado para compilar os módulos e produzir o executável. Os fontes serão compilados com o compilador gcc em um máquina virtual Linux. Os executáveis devem ser produzidos no mesmo diretório dos arquivos fontes O professor usará o GNU Make. Serão executadas (a partir dos scripts) o seguinte comando:
 - **make progr**
3. O programa será executado automaticamente várias vezes: uma vez para cada arquivo de testes e o resultado produzido será inspecionado visualmente pelo professor. Cada execução produzirá (pelo menos) um arquivo .svg diferente dentro do diretório informado na opção -o. Possivelmente serão produzidos outros arquivos .svg e .txt.

RESUMO DOS PARÂMETROS DO PROGRAMA

Parâmetro / argumento	Opcional	Descrição
-e <i>path</i>	S	Diretório-base de entrada (BED)
-f <i>arq.geo</i>	N	Arquivo com a descrição da cidade. Este arquivo deve estar sob o diretório BED .
-o <i>path</i>	N	Diretório-base de saída (BSD)
-q <i>arqcons.qry</i>	S	Arquivo com consultas. Este arquivo deve estar sob o diretório BED .
-ib	S	Inicia coleta de dados de desempenho
-cb	S	Continua a coletar dados
-fb <i>arq titulo</i>	S	Finaliza a coleta, produz no diretório BED o arquivo <i>arq.svg</i> contendo o gráfico com o título especificado
-ldd	S	(default) Usar lista duplamente encadeada com alocação dinâmica
-lse	S	Usar lista simplesmente encadeada com alocação estática

RESUMO DOS ARQUIVOS PRODUZIDOS

-f	-q	comando com sufixo	arquivos
<i>arq.geo</i>			<i>arq.svg</i>
<i>arq.geo</i>	<i>arqcons.qry</i>		<i>arq.svg</i> <i>arq-arqcons.svg</i> <i>arq-arqcons.txt</i>
<i>arq.geo</i>	<i>arqcons.qry</i>	<i>sufx</i>	<i>arq.svg</i> <i>arq-arqcons.svg</i> <i>arq-arqcons.txt</i> <i>arq-arqcons-sufx.[svg txt]</i> ⁷

ATENÇÃO:

- * os fontes devem ser compilados com a opção `-fstack-protector-all`.
- * adotamos o padrão C99. Usar a opção `-std=c99`.
- * compilar mantendo a tabela de símbolos do programa (opção `-g`).

APENDICE

<https://www.gnu.org/software/make/manual/make.html>
<http://opensourceforu.com/2012/06/gnu-make-in-detail-for-beginners/>

⁷ Podem ser produzidos os respectivos arquivos *.svg* e/ou *.txt*, dependendo da especificação do comando.