

1º trabalho de Física Computacional (2021-22, 1º semestre)

Entrega do trabalho

Até às 18H de dia 4 de Dezembro de 2021 (sábado) através do svn, **única e exclusivamente**.

Não se esqueçam de fazer `commit` de todos ficheiros com excepção dos ficheiros `*.o*.exe`

A operação `svn status` permite identificar os ficheiros ainda não `committed` ou que não estejam ainda sob controlo de svn.

Organização das pastas de trabalho

Em cada grupo, foi criada a pasta **trab01** que contém as seguintes pastas e ficheiros (não se esqueça de começar por fazer `svn update`):

<code>trab01/src</code>	[user classes]
<code>/main</code>	[main programs]
<code>/bin</code>	[object files, executables]
<code>/lib</code>	[user library: libFC.a]
<code>/rootANA</code>	[analysis macro if needed]
<code>Makefile</code>	[makefile exemplo]
<code>Folha_Respostas.pdf</code>	[folha de respostas]
<code>T1.pdf</code>	[enunciado do trabalho]

De notar:

- Os programas principais que realizarem devem estar na pasta `main/`
- As classes (ficheiros “header” e “source”) desenvolvidos pelo grupo e necessários à resolução deste trabalho, devem estar na pasta `src/`
- O `Makefile` a usar deve possuir as seguintes tarefas definidas:
 - `make clean`:
apagar os ficheiros objecto `bin/*.o` e executáveis `bin/*.exe` e a biblioteca `lib/libFC.a`
 - `make lib`:
produção da biblioteca do trabalho `libFC.a`
 - `make trab01`:
produz o executável do programa principal que será desenvolvido no trabalho

Correcção e Cotação

A resolução do trabalho implicará a entrega:

- dos códigos C++ realizados (classes e programa principal)

- da folha de respostas (ficheiro [Folha_Respostas.pdf](#)) preenchida onde constará a identificação dos elementos do grupo que terão participado na realização do trabalho
> o software PDFexpert (macOS) e Okular (linux) permitem a escrita no ficheiro pdf

A folha de respostas pode ser preenchida à mão ou no computador e deve ser submetida no svn. Caso seja preenchida à mão proceda à sua digitalização e junte ao svn.

Passos que serão seguidos na correcção do exercício:

- clareza e comentários do código C++ (10%)
- estrutura do código C++ implementada (15%)
- resultados (75%)

Os trabalhos terão que ser entregues até às 18H do dia 4 de Dezembro. Caso existam entregas posteriores às 18H05 (atrasos superiores a 5 minutos de tolerância), a regra de desconto em valores [0,20] que será aplicada à nota do trabalho será a seguinte:

$$D = T * 0.2$$

D : desconto em valores [0,20]

T : tempo de atraso de entrega em minutos, a partir das 18H05 (fim da tolerância)

Enunciado

Mapa de luz produzido por fonte pontual e isotrópica numa superfície plana

Uma fonte de luz pontual isotrópica encontra-se a uma distância $d = 1$ metro de um plano de dimensões 4×4 metros, de acordo com a figura abaixo. O fluxo de energia por unidade de tempo, conhecido como **fluxo radiante**, emitido pela fonte (Φ) é de 5×10^3 Watts.

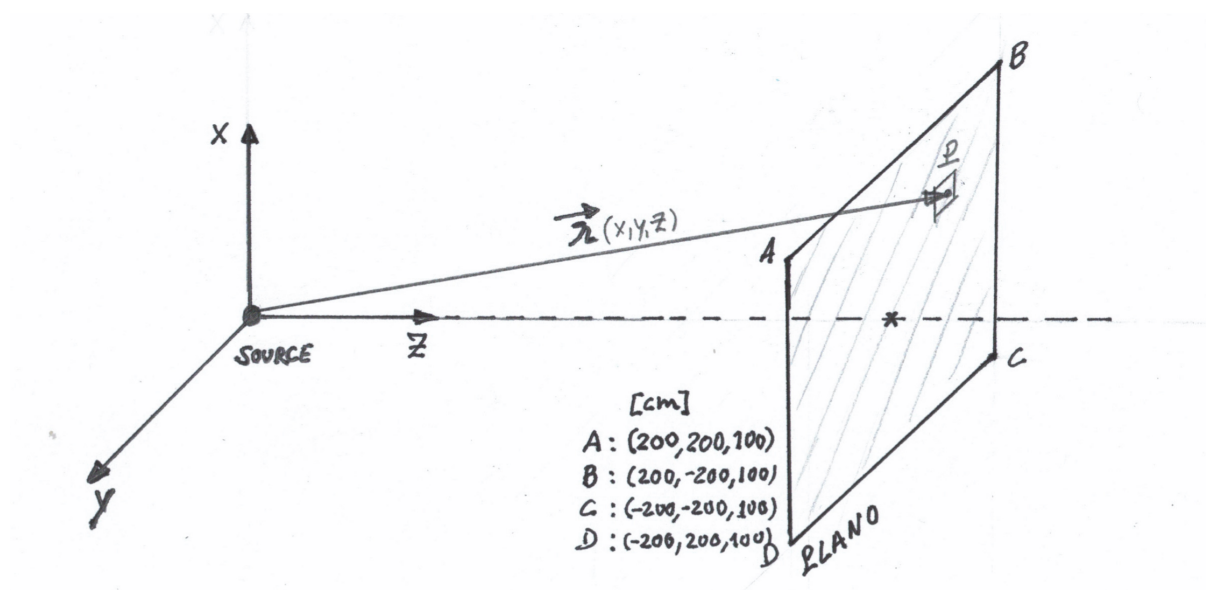


Figura 1: Geometria da fonte de luz e do plano iluminado

O objetivo do trabalho é produzir um mapa espacial que mostre a iluminação do plano e proceder aos cálculos pedidos nas alíneas que se seguem. Para tal irá desenvolver um programa `trab01_1.cpp` situado na pasta `trab01/main` e utilizar/desenvolver as classes e estruturas auxiliares que entender realizar e que deverão estar situadas na pasta `trab01/src`, que deverá produzir os seguinte resultados:

[13 valores]

- a. Determinar os valores da potência incidente por unidade de área do plano (Irradiância: $\frac{d\Phi}{dA}$, W/cm^2) nos seguintes pontos expressos em centímetros no sistema de coordenadas da figura:

X_1	X_2	X_3	X_4	X_5	X_6
(0, 0, 100)	(50, 0, 100)	(0, 50, 100)	(200, 100, 100)	(-100, 200, 100)	(60, 50, 100)

Imprimir de forma clara no ecrã os valores pedidos.

- b. Determinar o mapa de luz no plano. Para isso, assuma que o plano é dividido em células de largura $(\Delta x, \Delta y)$, integrando a função Irradiância em cada célula com a chamada regra do ponto médio,

$$P_{inc} = \int_{-200}^{200} \int_{-200}^{200} f(x, y) dx dy \rightarrow \sum_{i=1}^n \sum_{j=1}^m f(x_i, y_j) \Delta x \Delta y \quad (1)$$

onde:

- $x_i = x_{min} + (2i - 1) \frac{\Delta x}{2}$
- $y_j = y_{min} + (2j - 1) \frac{\Delta y}{2}$

Realizar um gráfico com o mapa do plano iluminado usando a paleta *kDarkBodyRadiator*,

```
#include "TStyle.h"
(...)
gStyle->SetPalette(kDarkBodyRadiator);
```

Produza o plot no ficheiro `trab01_mapa_1.pdf`

- c. Determine a potência total incidente no plano em Watts e obtenha uma estimativa do erro.

Imprima no ecrã o valor da potência total incidente no plano e do erro.

[3 valores]

- d. Faça um gráfico que mostre como varia a potência total incidente no plano em função da distância da fonte ao plano, entre 10 cm e 10 metros.

Produza o plot no ficheiro `trab01_mapa_2.pdf`

[4 valores]

- e. Pretende-se agora calcular o integral da potência incidente no plano de forma mais precisa e assim compararmos com o cálculo obtido anteriormente pela regra do ponto médio. Calcule assim o valor do integral numérico da potência incidente no plano com um erro inferior a 10^{-3} , usando os métodos de **simpson** e de **monte-carlo**.

$$P_{inc} = \int_{-200}^{200} \int_{-200}^{200} f(x, y) dx dy = \int_{-200}^{200} dy \int_{-200}^{200} f(x, y) dx \quad (2)$$

Imprima no ecrã os valores calculados para o integral numérico da potência incidente no plano e o seu erro, obtido por ambos os métodos.