

Lista de ejemplos para la utilización de wtpc.cls

Rodrigo Lugones

rodrigolugones@gmail.com

Ejemplos para utilizar la clase wtpc.cls

Por ahora, no hay demasiadas cosas raras para la utilización de esta clase de Beamer. Se utiliza como se utiliza Beamer, pero está simplificado el asunto de poner código. Además, la paleta de colores con la que se muestra el código es fácilmente modificable (por si no nos gusta o se ve mal en algún proyector).

Pasemos a ver unos ejemplos.

Incluir código desde un archivo

Para incluir código desde un archivo, utilizamos el paquete `lstlistings`. Si bien hay otros (`minted`, por ejemplo), este paquete es el más portable, ya que no requiere programas adicionales instalados en el sistema (`minted`, por ejemplo, requiere instalar `pygmentize`).

Entonces, para incluir código de, por ejemplo, C, ponemos:

```
1 /* Hello World program */
2 #include <stdio.h>
3 int main()
4 {
5     // printf() displays the string inside quotation
6     printf("Hello, World!");
7     return 0;
8 }
```

Incluir código desde un archivo

O si queremos incluir algo del *shell*, directamente:

```
#!/bin/sh  
echo "Hello world"
```

Incluir código *inline*

Si queremos introducir algo de código *inline*, es tan sencillo como poner `import numpy as np` en el caso de código propiamente dicho, o `for i in $(ls); do echo $i; done` para comandos de la consola.

Incluir código desde L^AT_EX

Si queremos incluir código desde L^AT_EX, es un poco más complicado. Hacerlo, requiere que la diapositiva (frame) sea iniciada con la opción [fragile] (hay un 'conflicto' con entorno verbatim, necesario para que L^AT_EX no intente interpretar en código que le estamos poniendo). De cualquier forma, si iniciamos la diapositiva con `\begin{frame}[fragile]` (y la cerramos con `\end{frame}`), todo anda bien.

Hecho esto, podemos incluir el código de la siguiente manera:

```
1 import numpy as np
2 class prueba(object):
3     def __init__(self, var1):
4         """clase prueba"""
5         self.variable = var1
```

Incluir código desde L^AT_EX

Cabe mencionar que si agregamos código así, tengamos cuidado, porque los espacios en blanco que escribamos los va a considerar. Por ejemplo, vean qué sucede si sigo la *indentación* que vengo usando en L^AT_EX:

```
1  import numpy as np
2  class prueba(object):
3      def __init__(self, var1):
4          """clase prueba"""
5          self.variable = var1
6
```

Noten no sólo que la *indentación* es más grande, sino también que los dos espacios antes de `\end{code}` los tomó como una línea de código extra. Conclusión: en general es más cómodo usar `\includecode` (o `\includeshell`).

Para agregar una consola:

```
#!/bin/sh  
echo ‘Hello world’
```

Cosas copadas de Istlisting

Hay algunas opciones de Istlisting que nos pueden venir bien.

- Poner determinadas líneas del código:

```
1  /* Hello World program */  
2  #include <stdio.h>  
3  int main()
```

- La magnífica posibilidad de resaltar líneas:

```
1  import numpy as np  
2  class prueba(object):  
3      def __init__(self, var1):  
4          """clase prueba"""  
5          self.variable = var1
```

Cosas copadas de Istlisting

Hay algunas opciones de Istlisting que nos pueden venir bien.

- Poner determinadas líneas del código:

```
1 /* Hello World program */
2 #include <stdio.h>
3 int main()
```

- La magnífica posibilidad de resaltar líneas:

```
1 import numpy as np
2 class prueba(object):
3     def __init__(self, var1):
4         """clase prueba"""
5         self.variable = var1
```

Cosas copadas de Istlisting

Hay algunas opciones de Istlisting que nos pueden venir bien.

- Poner determinadas líneas del código:

```
1 /* Hello World program */  
2 #include <stdio.h>  
3 int main()
```

- La magnífica posibilidad de resaltar líneas:

```
1 import numpy as np  
2 class prueba(object):  
3     def __init__(self, var1):  
4         """clase prueba"""  
5         self.variable = var1
```

Incluir figuras, etc

Para la inclusión de figuras, se sigue la misma lógica de siempre. Puede ser incluida con el entorno *figure*:

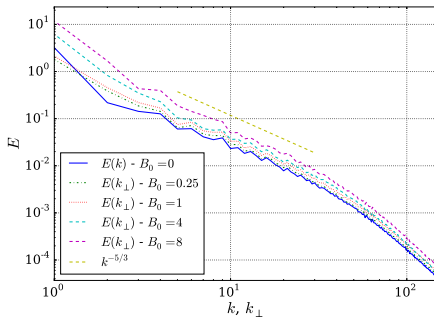
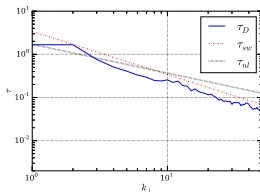


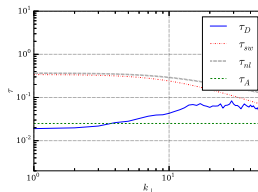
Figura: Reduced perpendicular energy spectra $E(k_{\perp})$ for the simulations with $B_0 = 0, 0.25, 1, 4$, and 8 , and isotropic energy spectrum $E(k)$ for the simulation with $B_0 = 0$. Kolmogorov scaling, $\sim k_{\perp}^{-5/3}$, is shown as reference.

Incluir figuras, etc

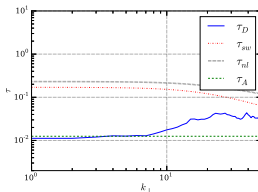
O con entorno *subfigure*:



(a) $k_{\parallel} = 0$



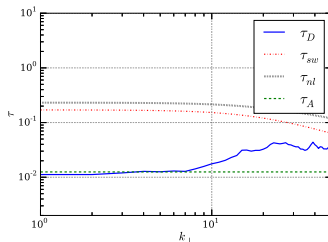
(b) $k_{\parallel} = 10$



(c) $k_{\parallel} = 20$

También lo podemos incluir más arcaicamente:

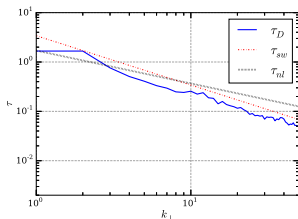
$$k_{\parallel} = 20$$



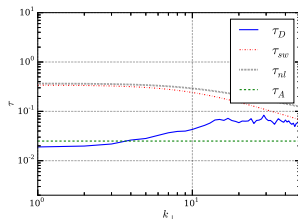
Incluir figuras, etc

Y es posible que necesitemos utilizar *minipages*:

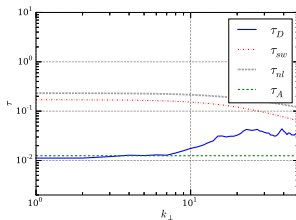
$$k_{\parallel} = 0$$



$$k_{\parallel} = 10$$



$$k_{\parallel} = 20$$



Lista de ejemplos para la utilización de wtpc.cls

Rodrigo Lugones

rodrigolugones@gmail.com