

Código simple para reconstruir la masa transversa del bosón W

Preámbulo:

Primero necesitas descargar el archivo con los datos, está en Google Drive y esta es la dirección para descargarlo:

["https://drive.google.com/file/d/0B74TqGKley_BUWVnSUtWQnFoSzA/view?usp=sharing"](https://drive.google.com/file/d/0B74TqGKley_BUWVnSUtWQnFoSzA/view?usp=sharing)

Una vez que lo hayas hecho crea un folder llamado "trabajo", y dentro de él crea los folders "myprogram1" y "samples", y dentro de "myprogram1" crea los folders "bin" y "lib".

Guarda el archivo que descargaste de Google drive en el folder llamado "samples".

Hecho esto entra a:

["https://github.com/Frigorifico9/Readding_Nero_Files/blob/master/myprogram1.cpp"](https://github.com/Frigorifico9/Readding_Nero_Files/blob/master/myprogram1.cpp), y asegúrate de que puedas ver en esta página el código de los archivos "makefile" y "myprogram1"

Requerimientos generales:

- 1.-Haber instalado Root
- 2.-Tener el folder previamente mencionado con el contenido indicado
- 3.-Usar un sistema operativo de Unix (Mac o Linux)

Requerimientos de la terminal:

Esto puede ser ligeramente diferente dependiendo de tu sistema operativo, como referencia yo uso "Mac OS X 10.11.5 64-bit".

- 1.-Abre tu terminal
- 2.-Introduce el siguiente código y da enter: `vi ~/.bashrc`
- 3.-Se abrirá dentro de la terminal tu "bash"
- 4.-Presiona la tecla de dirección hacia abajo hasta que ya no puedas bajar (no podrás llegar hasta abajo)
- 5.-Cuando ya no puedas bajar más copia y pega las siguientes instrucciones:

```
export ROOTSYS=/Users/Fer/root/root-6.06.06
source $ROOTSYS/bin/thisroot.sh
export PATH=$ROOTSYS/bin:$PATH
```

Nota que en la primera instrucción necesitas especificar la dirección de donde está Root en tu computadora

- 6.-Finalmente tienes que guardar estos cambios, para esto escribe `":q"` y presiona la tecla enter

Instrucciones:

1.-Copia el código que puedes ver en GitHub cuando das clic en “makefile” y pégalo en un archivo nuevo de un procesador de textos como TextEdit, Sublime Text 2, NotePad, o similares y guárdalo con el nombre “makefile” (tiene que ser este nombre) y como un archivo de texto, con extensión .txt por ejemplo, y guárdalo en el folder “myprogram1”

1.5.-Copia el código que puedes ver en GitHub cuando das clic en “myprogram1” y pégalo en un archivo nuevo de un procesador de textos como TextEdit, Sublime Text 2, NotePad, o similares

2.-En la línea 41 donde se carga el archivo entre las comillas tienes que poner la dirección del archivo que descargaste de Google Drive en tu computadora, por ejemplo en mi caso es: “/Users/Fer/Documents/trabajo/samples/NeroNtuples_9.root”
en tu caso debería de ser similar desde “/trabajo/samples/NeroNtuples_9.root”

3.-Guardar este archivo en el folder “myprogram1” con la extensión .cpp y con el nombre “myprogram1”, tiene que ser ese nombre por razones que se explicarán más adelante

4.-Abrir la terminal, esto se hace diferente en diferentes sistemas operativos. Si no sabes como usar la terminal a mi me sirvió mucho esta página:
“<http://www.macworld.co.uk/feature/mac-software/get-more-out-of-os-x-terminal-3608274/>”

5.-Usa la terminal de tu sistema operativo para llegar al folder “miprogram1”, si usas OS o Linux, al escribir “ls” en tu terminal deberías de poder ver en este momento los archivos contenidos en el folder “miprogram1”, los folders “bin” y “lib”, el archivo “makefile”, y ahora también “myprogram1.cpp”.

Si ves el folder y no únicamente sus archivos tienes que entrar al folder

6.-Tienes que activar el archivo “makefile” que debe de estar en el folder “miprogram1”.
Para esto si tienes OS o Linux solo escribe “make” en tu terminal

Nota sobre el “makefile”:

Este archivo en una “caja negra”, no importa que no sepas como funciona, sino como usarlo.

Este archivo lee códigos de root, y c++ en general, y construye con lo que leyó algo que la computadora puede entender, se conocen como “ejecutables” y suelen ser .exec, .exe, o .dmg.

Hacer este proceso se conoce como “compilar”.

Este archivo tiene una parte en la que dice que el archivo al que debe de convertir en un ejecutable se llama “myprogram1” por eso es importante guardar el código con ese nombre, pero eres libre de cambiar esa línea de texto y hacer que compile programas con otro nombre si quieres

Esta es la parte del “makefile” que cambiarías, solo la parte del final:

```
#####
###
# Build main.c compiling only the first prereq: ($<) means main.c
#####
###
myprogram1 : myprogram1.o
```

7.-Si después de unos momentos tras haber escrito el comando “make” ves algo como:
***g++ -g -Wall -fPIC -Wno-deprecated -ggdb -pthread -stdlib=libc++ -std=c++11 -m64
 -I/Users/Fer/root/root-6.06.06/include -laux -c -o myprogram1.o myprogram1.cpp
 compiling and linking myprogram1***

Significa que no hubo problemas, específicamente la parte de “***compiling and linking myprogram1***”

7.5.-Si ves algún mensaje de error en tu terminal: verifica que la dirección del archivo es correcta en la línea 41, y que dentro del folder “myprogram1” tengas los folders “lib” y “bin”, también verifica que no hayas hecho cambios en el “makefile”

8.-Si tu archivo se compiló sin problemas ahora puedes correrlo escribiendo en la terminal “bin/myprogram1”.

Si todo sale bien verás esto en tu terminal: “Info in <TCanvas::Print>: pdf file masaW.pdf has been created”

9.-Si vas al folder “myprogram1” en este deberías de encontrar el archivo pdf “masaW.pdf” que debería de ser una gráfica con valor máximo cerca de la masa conocida del bosón W de 80 GeV

En caso de dudas esta es mi información de contacto:

francofelix94@gmail.com

Facebook: Fernando Franco Félix