1. Fluent UDF tutorial.

An airfoil in a free shear layer (a wake, or due to the stratification of the atmosphere).

Design the inlet conditions for this case. Then apply this inlet condition for a computation of an airfoil (you can use your coursework airfoil, but don't include today's work into the coursework report).



Copy the uvinlet_tut1.c from the link in a text editor,

http://www.soton.ac.uk/~zxie/SESS6021/Airfoil/UDF/uvinlet_tut1.c

and save it in the folder where the Fluent case file is located. You must modify the uvinlet_tut1.c to suit your application.

For your refs, I have inserted more comments in the code as follows (also you may find more introductions here,

http://hpce.iitm.ac.in/website/Manuals/Fluent_6.3/Fluent.Inc/fluent6.3/help/html/udf/node230.htm).

uvinlet_tut1.c

UDF for specifying steady-state velocity profile boundary condition

```
#include "udf.h"
```

DEFINE_PROFILE(inlet_x_velocity, thread, position) /* function inlet_x_velocity, thread is a pointer to the face's thread, position is an integer that is a numerical label for the variable being set within each loop */

{

```
real x[ND_ND]; /* this will hold the position vector, ND_ND is defined as 2 for

FLUENT 2D) and 3 forFLUENT 3D*/

real y;

face_t f; /* declare f as a face_t data type */
```

begin_f_loop(f, thread) /*to loop over each face in the zone to create a profile*/

```
{
      F_CENTROID(x,f,thread); /*Within each loop, F_CENTROID outputs the value of the face
                              centroid (array x) for the face with index f that is on the thread
                                                    pointed to by thread. */
      y = x[1]; /*x[1] is assigned to variable y; note x[0] is for streamwise coordinate */
      F_PROFILE(f, thread, position) = (10 + y/12.5)*cos(5.0/180.*3.1415927); /*F_PROFILE uses
the integer position (passed to it by the solver based on your selection of the UDF as the boundary
condition for Evelocity in the Velocity Inlet panel) to set the E velocity in memory Assuming
domain is from y = -12.5m to y = +12.5m. Angle of attack is 5 degree.*/
  end f loop(f, thread)
}
/* Following the above, define the 'inlet_y_velocity' for the y velocity
at the inlet. */
DEFINE_PROFILE(inlet_y_velocity, thread, position)
{
/*you must code here by yourself. */
}
```

Open the Interpreted UDFs panel.

Define \rightarrow User-Defined \rightarrow Functions \rightarrow Interpreted...

- Interpreted UDFs -		
Source File Name		
/nfs/homeserver/ho	Browse	
CPP Command Name		
cpp		
Stack Size		
10000		
📕 Display Assembly Listing		
Use Contributed C	PP	
Interpret Clos	se Help	

In the **Interpreted UDFs** panel, select the UDF source file(s) you want to interpret by either typing the complete path in the **Source File Name** field or use the browser.

In the **Interpreted UDFs** panel, specify the C preprocessor to be used in the **CPP Command Name** field. You can keep the default cpp or you can select **Use Contributed CPP** to use the preprocessor supplied by Fluent Inc. Keep the default **Stack Size** setting of 10000, unless the number of local variables in your function will cause the stack to overflow. In this case, set the **Stack Size** to a number that is greater than the number of local variables used.

Keep the **Display Assembly Listing** option on if you want a listing of assembly language code to appear in your console window when the function interprets. This option will be saved in your case file, so that when you read the case in a subsequent **FLUENT** session, the assembly code will be automatically displayed.

Click Interpret to interpret your UDF.

!! Note that if your compilation is unsuccessful, then **FLUENT** will report an error and you will need to debug your program.

Close the Interpreted UDFs panel when the interpreter has finished.

Write the case file if you want the interpreted function(s) (e.g., inlet_x_velocity) to be saved with the case, and *automatically* interpreted when the case is subsequently read. If the **Display Assembly Listing** option was chosen, then the assembly code will appear in the console window.

To hook the UDF to FLUENT as the velocity boundary condition for the zone of choice, open the Velocity Inlet panel .

-	Velocity Inlet		
Zone Name			
velocity-inlet-11			
Velocity Specification Method	Components		V
Reference Frame	Absolute		▼
X-Velocity (m/s)	٥	udf inlet_x_velocity	V
Y-Velocity (m/s)	0	constant	V
Temperature (k)	300	constant	V
Outflow Gauge Pressure (pascal)	90000	constant	V
ОК	Cancel Hel	p	

Define → Boundary Conditions...

Note also to hook the inlet_y_velocity if applicable.

Run the Fluent solver and compare the results with those imposing constant velocity at inlet. (OK, for your convenience, this is the final UDF file for defining u & v at the inlet BCs, http://www.soton.ac.uk/~zxie/SESS6021/Airfoil/UDF/uvinlet_tut2.c)