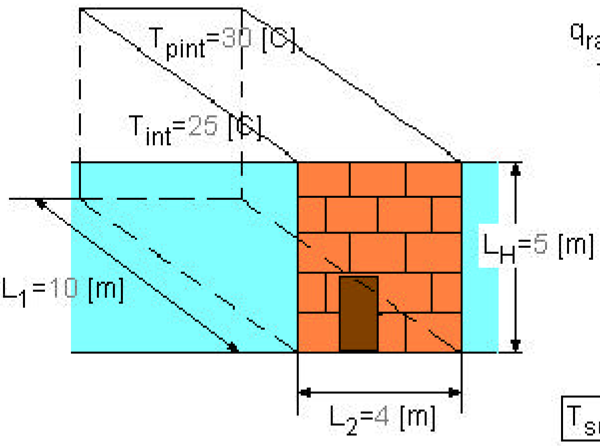
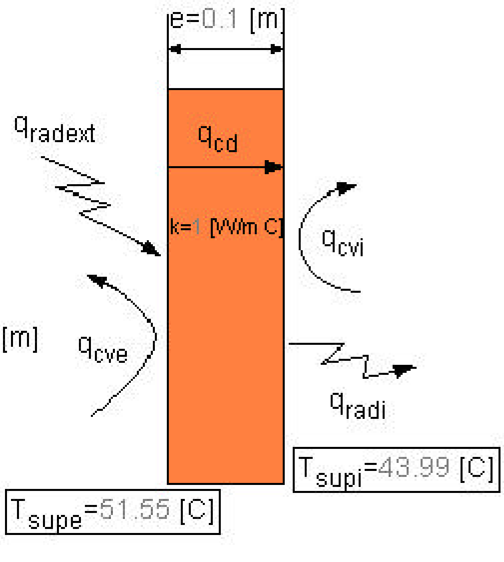
Colección De Problemas De Transferencia De Calor Mediante EES





Índice

1. Introducción

Aquí se presenta una colección de problemas donde se ponen de manifiesto todos los mecanismos de transferencia, conducción, convección y radiación. En la resolución de estos problemas se ha pretendido conjugar todas las posibilidades que ofrece el programa EES. Las aplicaciones más usuales son las siguientes:

- Uso de **Diagram Window**, utilizado para representar esquemáticamente el problema, ayudando a un mejor entendimiento del mismo y a plantear las ecuaciones para su solución.

- Uso de las **Thermophysical Property Functions** (funciones de propiedades termofísicas). Esto es muy importante, ya que nos permite poner las propiedades termofísicas (viscosidad, conductividad, número de Prandlt...) en función de variables como la temperatura, y poder resolver el problema sin iterar, haciéndolo el propio programa.

- Uso de las **User library routines** (funciones de líbrería) y **Modules**. Permite usar todas las funciones desarrolladas por el usuario, y en este caso funciones relacionadas con todas las aplicaciones de la transferencia de calor, como las correlaciones, factores de forma, ábacos...

- Uso de **Parametric Table Window**. Esta opción nos permite resolver el problema tantas veces como queramos en función de una o más variables del problema. Los valores de las variables paramétricas se introducen en una tabla, y el programa resuelve el problema para cada valor diferente de estas variables.

- Uso de **Plot Windows**. Las distintas solucionés obtenidas con la tabla paramétrica podemos representarlas gráficamente mediante este comando. Tenemos que elegir las variables que queremos representar y el programa nos proporciona la curva, y otras opciones para modificar ejes, ponerle nombre al gráfico...

En el siguiente punto, guía práctica, se dará más detalles del uso de estas herramientas, además de explicar también casi lo más importante, las estimaciones iniciales de las variables, ya que eso es fundamental para que el problema converja y nos de una buena solución.

La colección consta de 21 problemas, alguno de ellos con subapartados de un problema principal, y se encuentran repartidos entre los siguientes temas:

- Conducción en régimen permanente.
- Conducción en régimen transitorio.
- Aletas.
- Convección.
- Radiación.

- Mecanismos combinados.
- Intercambiadores de calor.

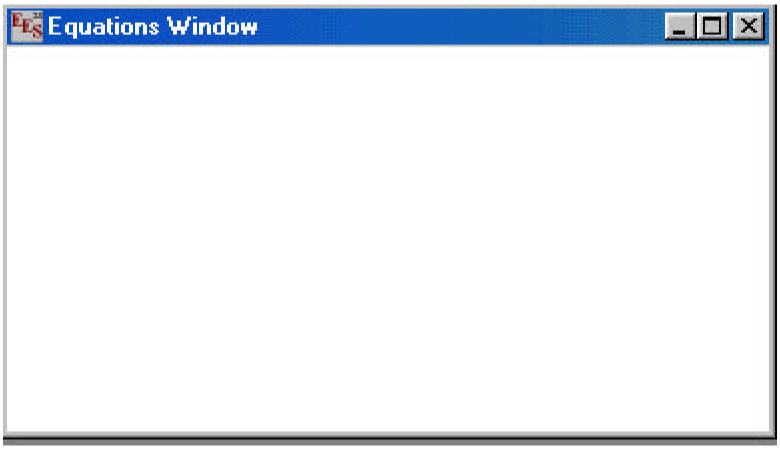
2. *Guía Práctica*

Basándonos en problemas de la colección se va a explicar cómo se utilizan las herramientas mencionadas en la introducción. No van a ser vistas con demasiado detalle puesto que existe un manual de usuario del programa EES donde se tratan con más profundidad, pero sí lo suficiente para tener unas ideas básicas.

Los apartados que veremos son:
- **Equations Window** (ventana de ecuaciones).
- **Unit System** (sistema de unidades).
- **Thermophysical Property Functions** (funciones de propiedades termofísicas).
- **User library routines** (funciones de líbrería) y **Modules**.
- **Variable Info** (estimaciones iniciales).
- **Parametric Table Window** (tabla paramétrica).
- **Plot Windows** (representaciones gráficas).
- **Diagram Window.**

Equations Window

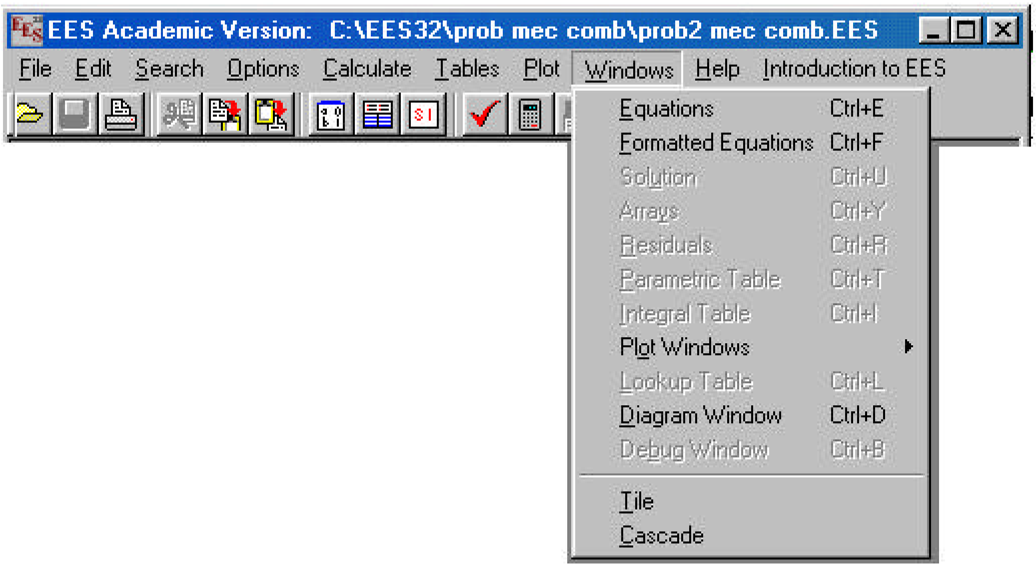
Cuando abrimos un archivo nuevo nos aparece la ventana de ecuaciones (**Equations window**), y ahí es donde escribimos las ecuaciones para la resolución del problema.



También podemos escribir texto si lo encerramos entre llaves, o entre comillas dobles. Si es entre comillas dobles el texto aparece también en **Formatted Equations Window**, donde se presentan las ecuaciones en formato matemático.

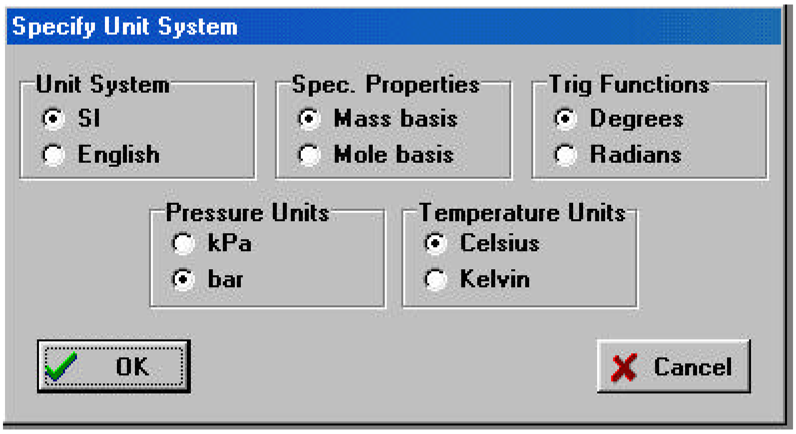
Pueden utilizarse los comandos de edición, cortar, copiar, pegar, que se encuentran en el menú **Edit**, de manera normal a cualquier procesador de texto.

Para hacer activa una ventana o para hacerla visible si no estuviera en pantalla, picamos en el menú **Windows**, y se despliega una persiana, como se muestra en el dibujo, con todas las ventanas que podemos activar.



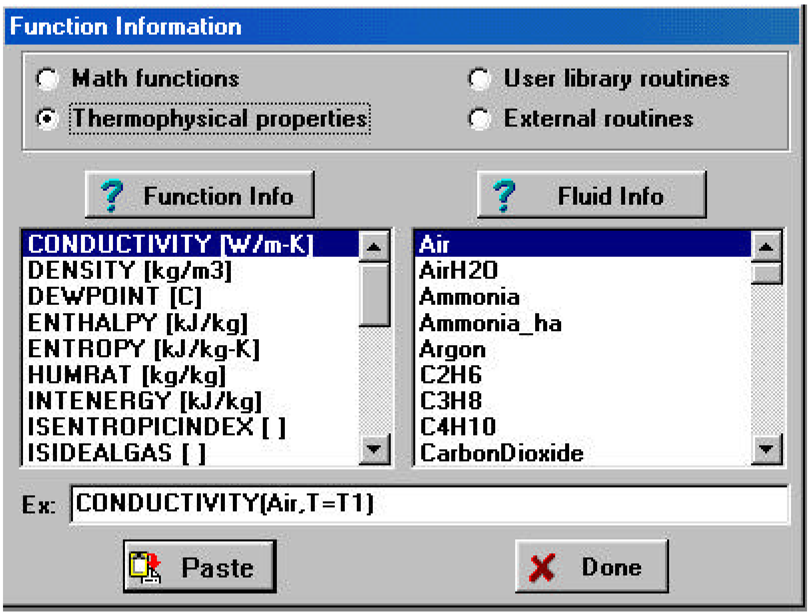
Unit System

El comando **Unit System** en el menú **Options** nos proporciona una ventana donde nos indica las unidades de las variables usadas por las funciones matemáticas y de propiedades termofísicas. Es muy importante establecer las unidades, ya que si utilizamos una función trigonométrica, o de propiedades como por ejemplo la densidad, o alguna librería interna, pueden dar resultados erróneos. Si queremos establecer los valores para nuestro problema actual basta con picar el botón **OK**.



Thermophysical Property Functions

Para utilizar una función de propiedades termofísicas picamos en el comando **Function Info** del menú **Options,** y nos aparecerá el siguiente cuadro de diálogo.

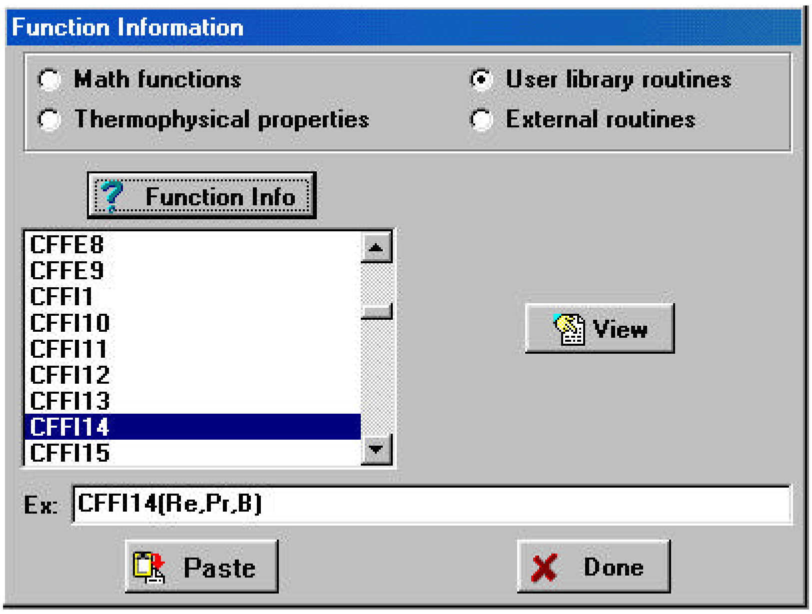


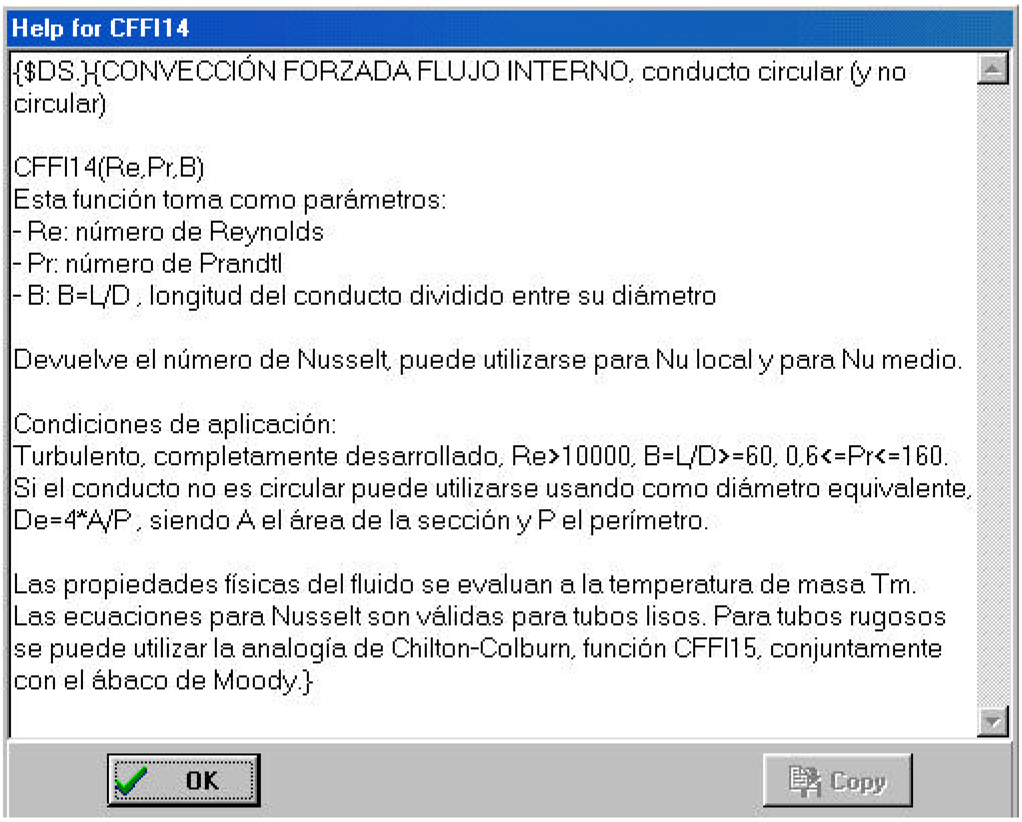
Los cuatro botones superiores nos indican cuál es la información que vamos a obtener, en este caso picamos **Thermophysical properties**. Como se puede ver, en la lista de la izquierda aparecen las propiedades indicando las unidades en las vienen expresadas (eso lo elegimos con **Unit System**, como vimos antes), y en la lista de la derecha la sustancia. Disponemos de dos botones de información, de la función y del fluido, y picando ahí podemos obtener una ayuda sobre sus características.

Picando en el botón **Paste** llevamos la función a la ventana de ecuaciones en la posición donde tuviéramos el cursor.

User library routines y Modules

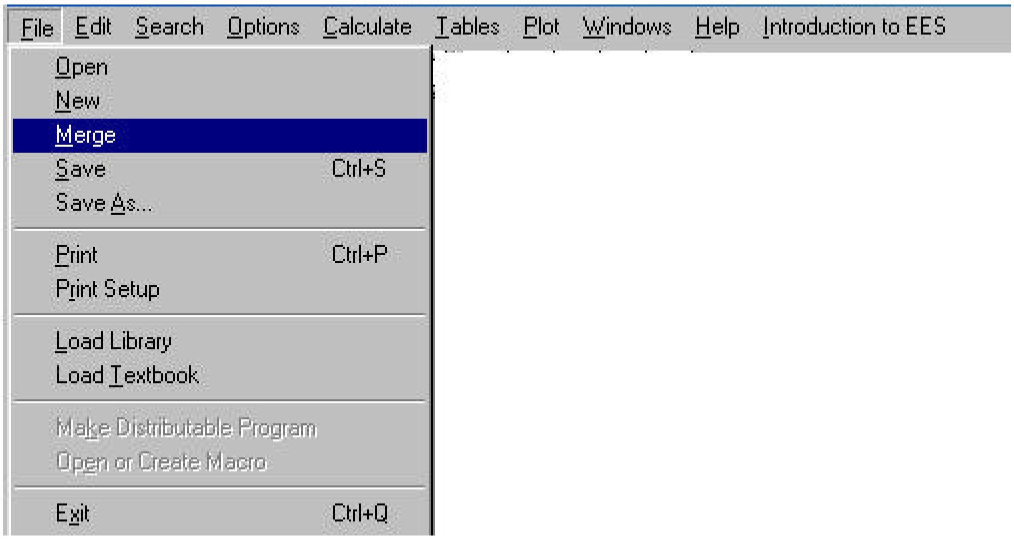
Para utilizar las funciones de la librería se procede de la misma manera que para las funciones termofísicas, pero picando en **User library routines**, y nos aparece un cuadro muy similar al anterior.
Aparece una lista donde podemos elegir la función que necesitamos, y además tenemos un botón de ayuda, **Function Info**, y otro botón, **View**, donde se puede ver el listado de la función. Como ejemplo vamos a ver la información de la función CFFI14.



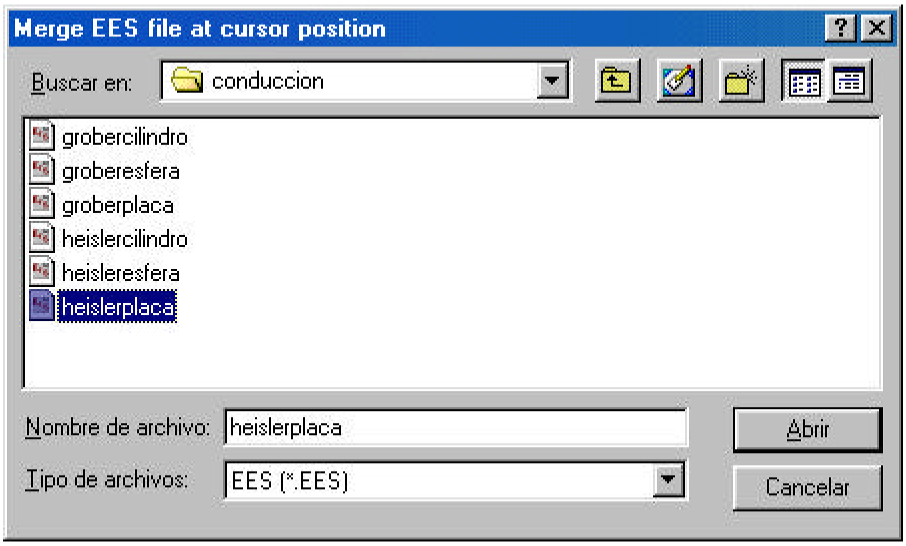


Picando en el botón **Paste** llevamos la función a la ventana de ecuaciones en la posición donde tuviéramos el cursor.

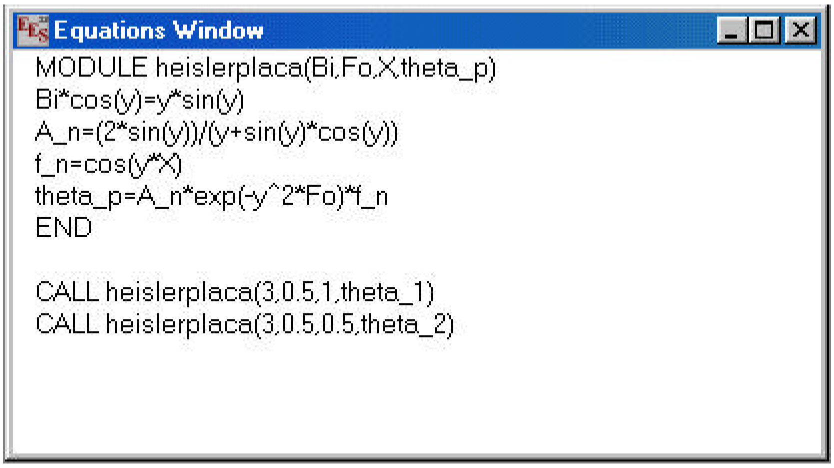
Los **Modules** son subprogramas que puede realizar el usuario e incorporarlo a la resolución de su problema en **Equation Window.** En las aplicaciones realizadas para la transferencia de calor existen **Modules** para los ábacos de Heisler y de Gröber, para las distintas geometrías de placa plana, cilindro y esfera. Es conveniente insertar el subprograma al principio de las ecuaciones. Si vamos a utilizar las relaciones de Heisler o Göber se introduce el subprograma al principio de todas las ecuaciones con el comando **Merge**, y cada vez que necesitemos usar el subprograma lo podemos llamar tantas veces como queramos y con valores diferentes cada vez. Vamos a ver como se haría. Vamos a partir de un problema nuevo y tenemos abierta nuestra ventana de ecuaciones. Vamos al menú **File**, y picamos en el comando **Merge**.



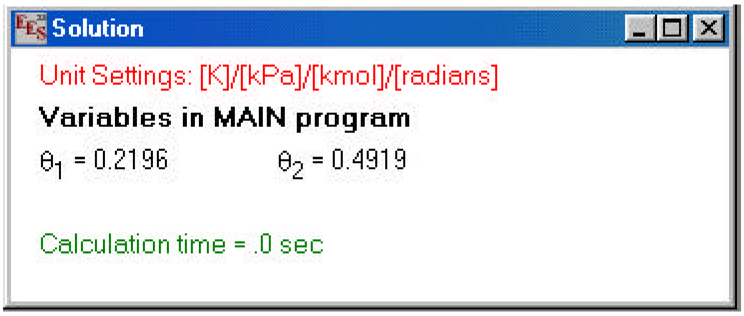
Nos aparece una ventana donde elegimos el fichero desde el que queremos insertar nuestro subprograma, en nuestro caso iremos a la carpeta **conduccion**, dentro de la carpeta **Userlib**, y elegiremos el archivo **heislerplaca**.



El subprograma se insertará en la posición donde tengamos el cursor en la ventana de ecuaciones, y una vez instalado podemos llamarlo cuantas veces queramos con los valores que queramos, devolviendo el valor a la variable de salida que indiquemos. Veamos un ejemplo.

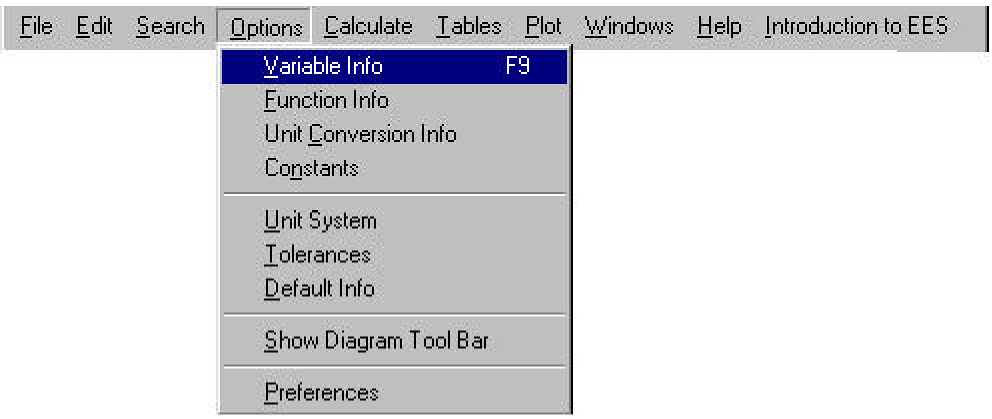


Arriba tenemos el subprograma y después hemos realizado dos llamadas, las primera nos devolverá el valor de $theta_1$ en el punto 0.5 de la placa, y la segunda llamada nos devolverá el valor de $theta_2$ en la superficie de la placa (X=1). Picamos en **Solve** en el menú **Calculate**, y obtenemos la solución.



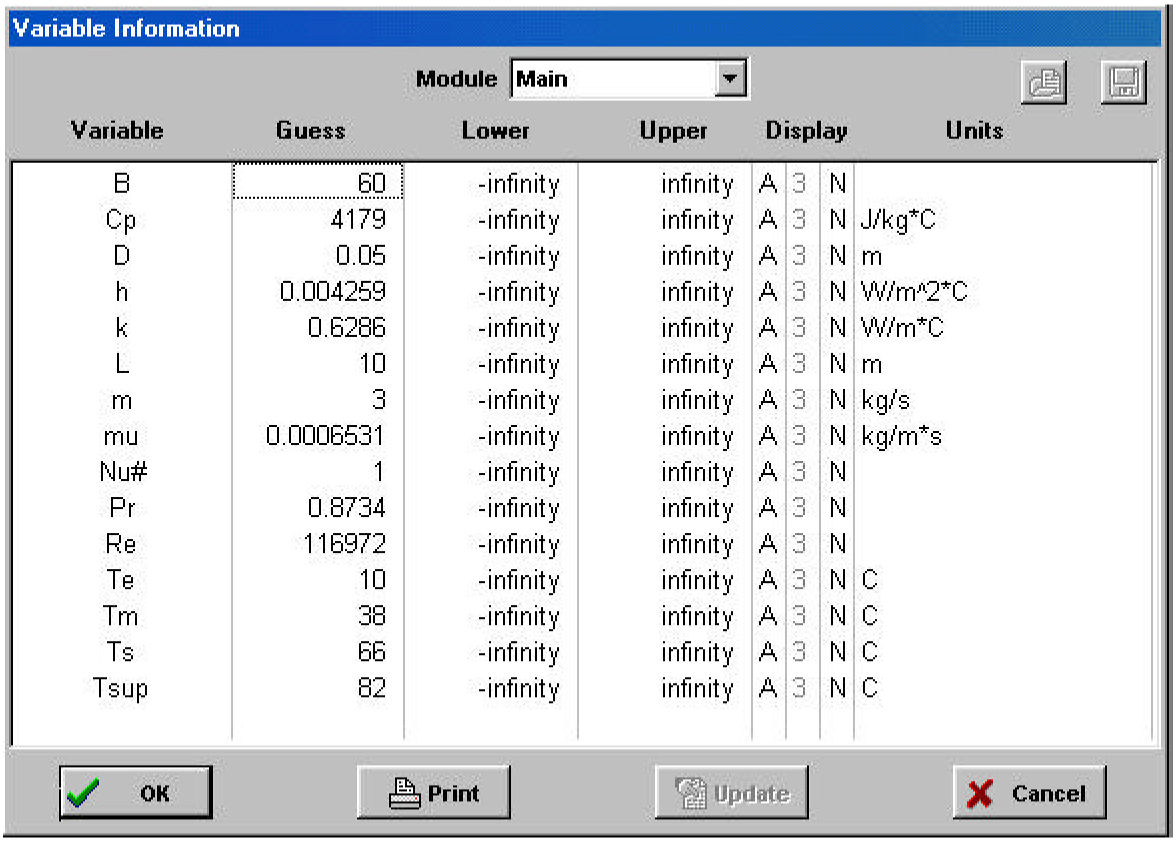
Las variables locales no aparecen a menos que lo seleccionemos en el comando **Preferences** en el menú **Options**.

Variable Info



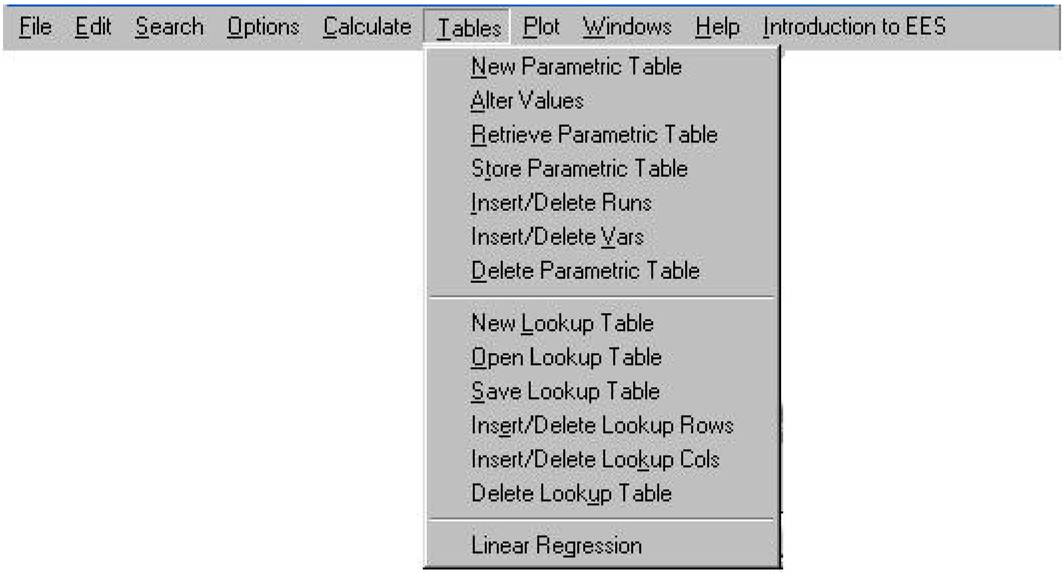
Este comando nos ofrece una ventana donde aparecen todas las variables del problema con una columna donde debemos darle unos valores iniciales. Es una de las cosas más importante a la hora de solucionar un problema mediante un programa. Si no estimamos bien estos valores puede que el problema no converja o no nos proporcione una correcta solución. Esto se hace todavía más importante si cabe con los problemas de transferencia de calor, donde los valores de los números adimensionales como Re, Ra, Gz tienen que estar dentro de un rango para la validez de la correlación, y donde tenemos que partir de buenas aproximaciones de los coeficientes de película para calcular las temperaturas superficiales y seguir con las iteraciones. El programa es el que itera, y a una velocidad infinitamente superior a la nuestra, pero si tiene unos buenos valores iniciales. Las funciones de librería de correlaciones tienen restringido los límites superior e inferior de Re, o Ra, mostrándonos un mensaje de error si esos parámetros no están dentro de su rango. Esto nos obliga a entrar siempre en esta ventana y darle valores a nuestras variables, y cuanto más cercanos a los reales menos problemas habrá para que el programa encuentre la solución.

En esta ventana podemos también ponerle límites superior e inferior a las variables, y ponerle las unidades. El programa no entiende de unidades, excepto cuando se utilizan las funciones trigonométricas o de propiedades, luego tenemos que ser coherentes con todos los valores de nuestras variables.

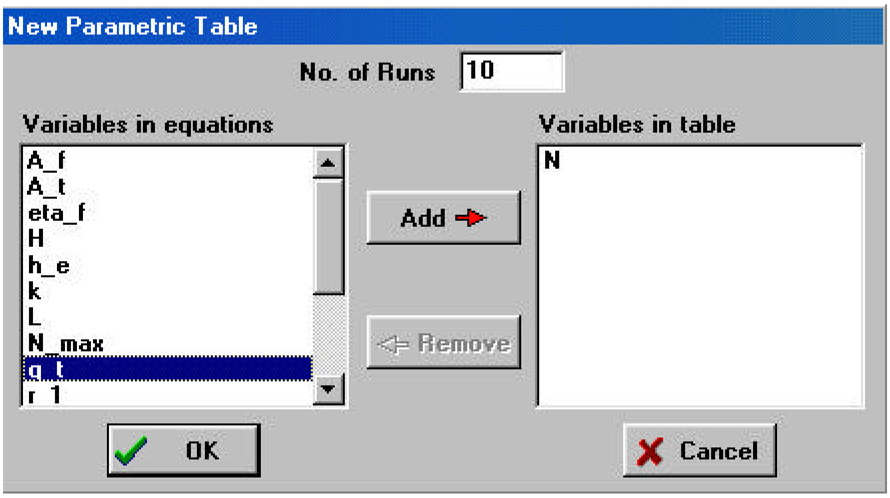


Variable	Guess	Lower	Upper	Display			Units
B	60	-infinity	infinity	A	3	N	
Cp	4179	-infinity	infinity	A	3	N	J/kg*C
D	0.05	-infinity	infinity	A	3	N	m
h	0.004259	-infinity	infinity	A	3	N	W/m^2*C
k	0.6286	-infinity	infinity	A	3	N	W/m*C
L	10	-infinity	infinity	A	3	N	m
m	3	-infinity	infinity	A	3	N	kg/s
mu	0.0006531	-infinity	infinity	A	3	N	kg/m*s
Nu#	1	-infinity	infinity	A	3	N	
Pr	0.8734	-infinity	infinity	A	3	N	
Re	116972	-infinity	infinity	A	3	N	
Te	10	-infinity	infinity	A	3	N	C
Tm	38	-infinity	infinity	A	3	N	C
Ts	66	-infinity	infinity	A	3	N	C
Tsup	82	-infinity	infinity	A	3	N	C

Parametric Table Window

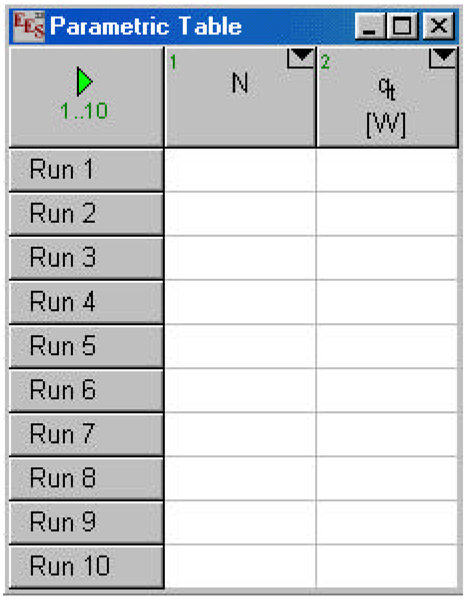


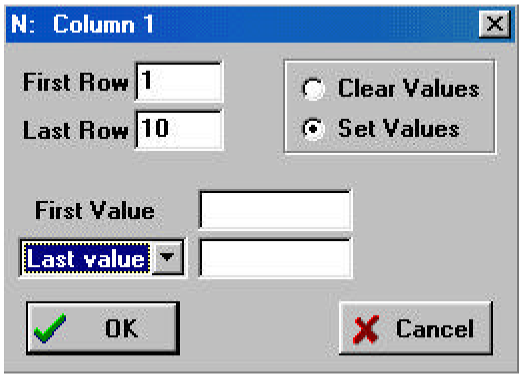
En el menú **Tables** encontramos todos los comandos relacionados con las tabla paramétrica. **Parametric Table** se usa para automatizar cálculos repetitivos, para resolver ecuaciones diferenciales, para representar los datos en un gráfico, y para ajustar datos a una curva. Vamos a ver los pasos que seguimos para realizar una tabla y posteriormente veremos cómo podemos representar gráficamente las variables. Primero picamos en **New Parametric Table**, en el menú **Tables** y nos aparecerá una ventana donde elegimos las variables que queremos que formen la tabla, y el número veces que va a calcular la solución para los distintos valores.



La variable paramétrica, que es a la que vamos a darle valores, no puede estar definida en la ventana de ecuaciones porque si no nos saldría un mensaje de error. Podemos borrarla, o simplemente encerrarla entre paréntesis, con lo que el programa no la reconoce como variable sino como comentario.

En la tabla, picando en el triangulito negro superior derecha que hay en la celda de la variable nos aparece una ventanita donde asignamos valores a la variable, como se muestra en la siguiente figura.





Una vez que hemos asignados los valores, picamos en el triangulito verde para resolver, obteniendo los resultados en la columna correspondiente.

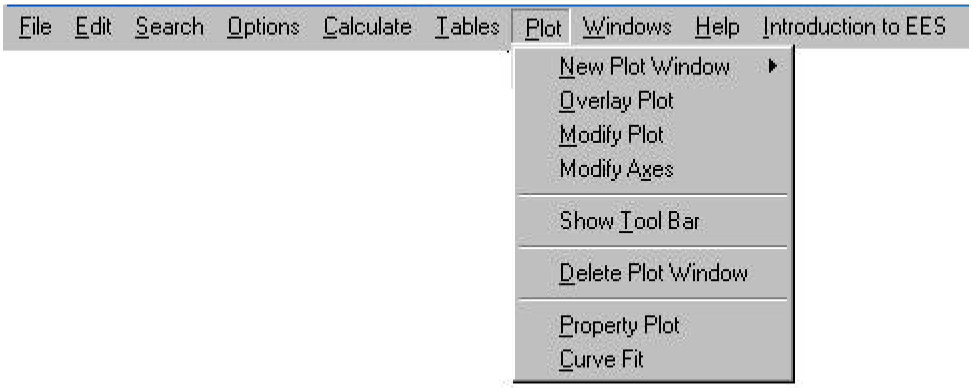
Parametric Table

1..10	N	q_t [W]
Run 1	6	798.5
Run 2	7	892.3
Run 3	8	986.1
Run 4	9	1080
Run 5	10	1174
Run 6	11	1267
Run 7	12	1361
Run 8	13	1455
Run 9	14	1549
Run 10	15	1643

Existen comandos para alterar los valores, insertar más filas, insertar más variables, etc, explicados con detalles en el manual del programa EES.

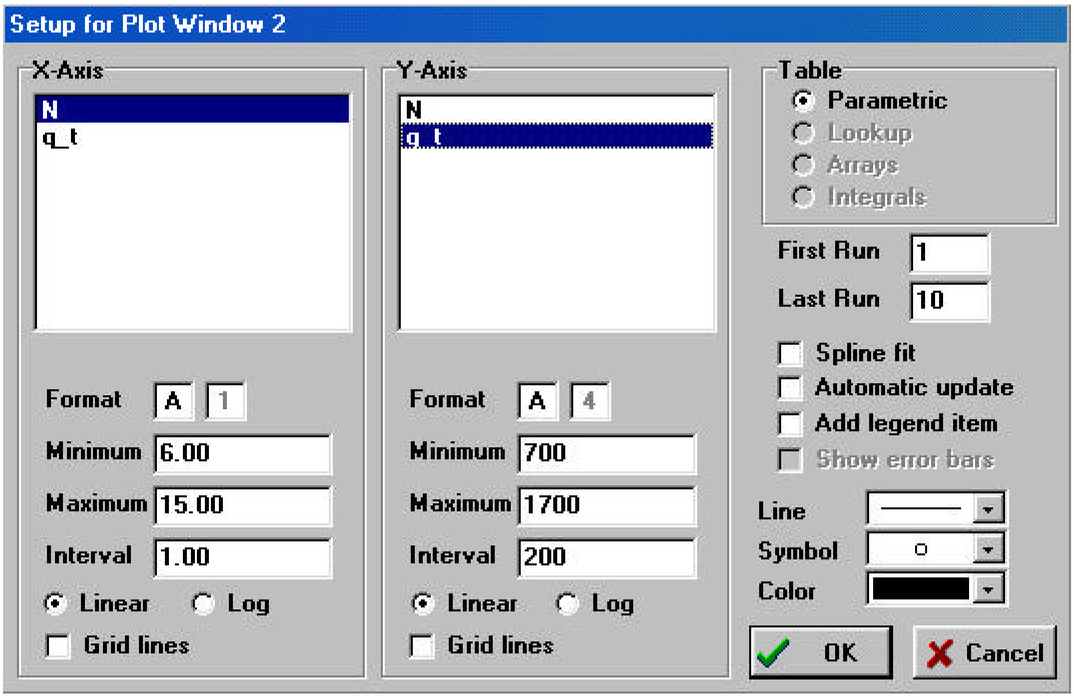
Estos análisis paramétricos son muy interesantes porque junto con su representación gráfica que veremos enseguida dan una idea muy intuitiva de cómo varían unas magnitudes frente a otra de una manera muy fácil y rápida.

Plot Windows



En el menú **Plot** encontramos los comandos para realizar representaciones gráficas. Nos permite dibujar las variables definidas en la **Parametric Tables**, **Lookup Table** o **Arrays Tables** en función de otras variables en esa tabla.

Veamos qué pasos hay que dar para representar en un gráfico un variable frente a otra. Empezamos picando en **New Plot Window**, elegimos qué tipo de representación queremos y nos aparece una ventana donde elegimos las variables de cada eje, y una serie de posibilidades más que nos ofrece.



Pulsando **OK** tendremos la representación que queríamos. Picando en **Show Tool Bar** tenemos una barra de herramientas para escribir sobre el gráfico, añadir líneas, figuras...

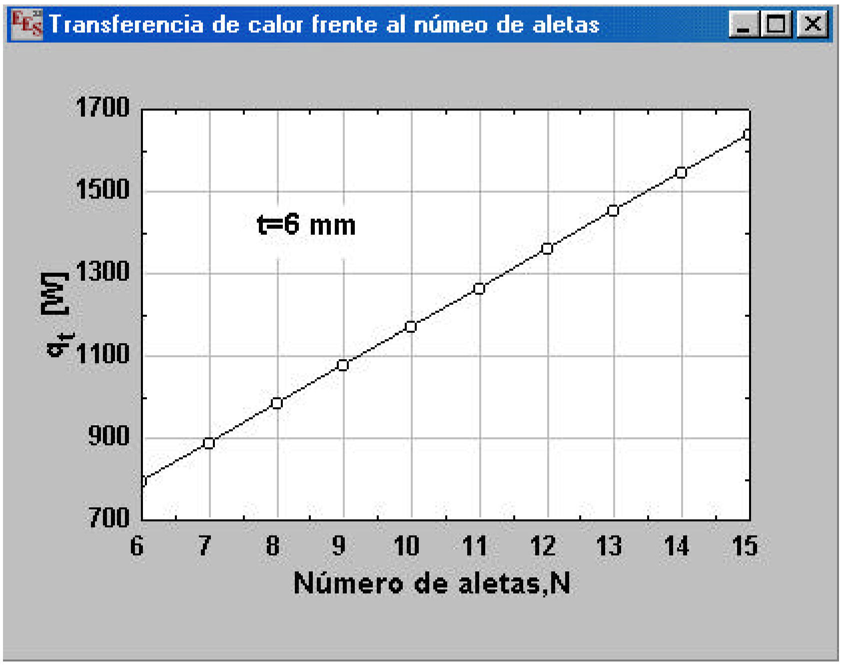
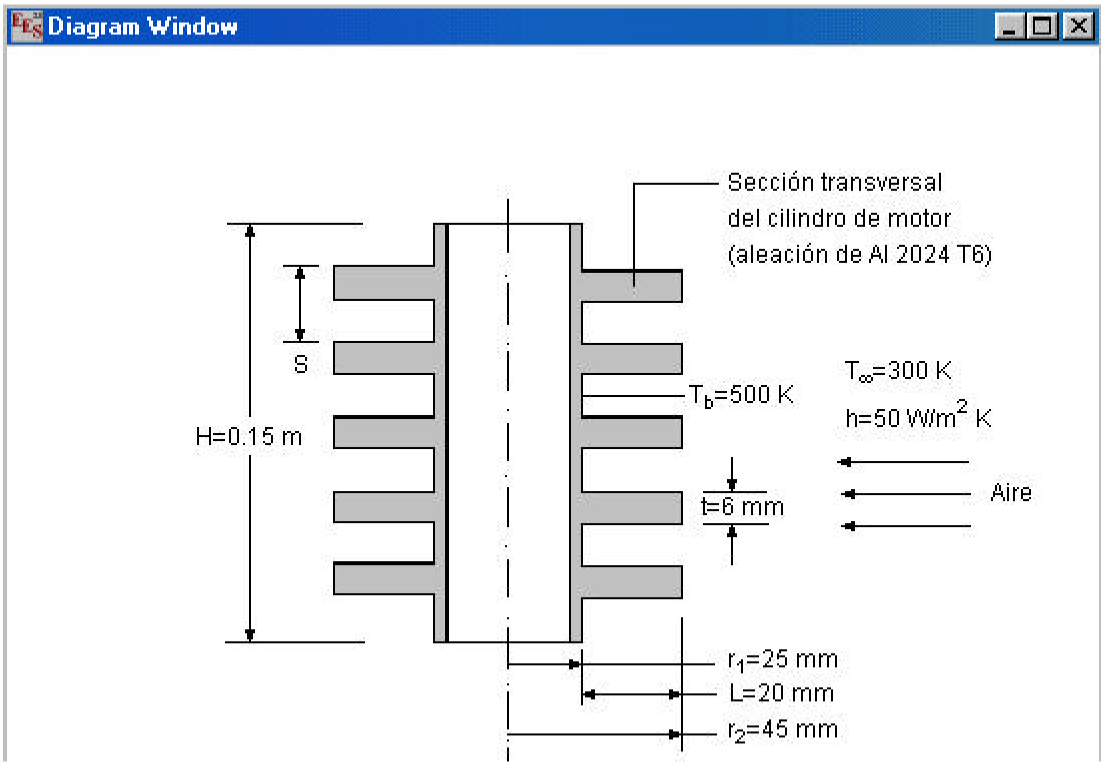


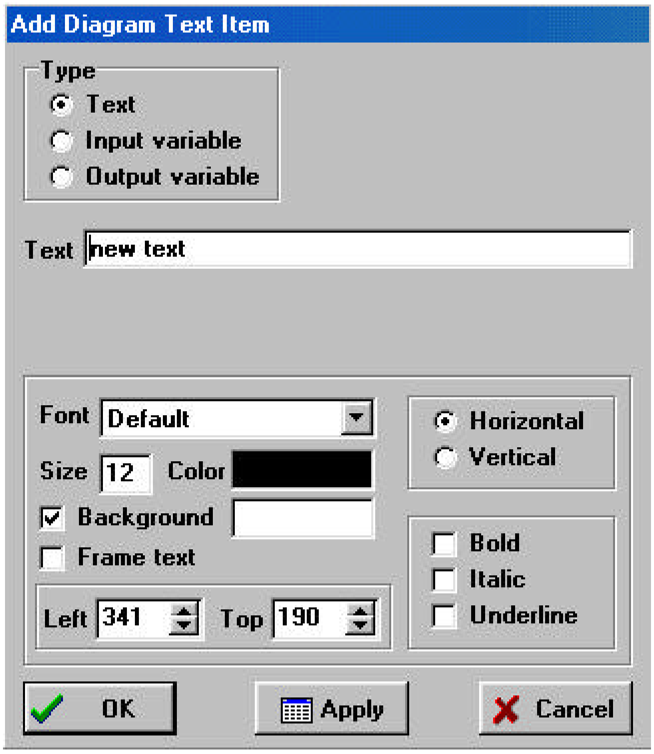
Diagram Window

El programa EES nos permite realizar esquemas y dibujos en **Diagram Window**, para así poder mostrar más claramente el problema o ayudarnos a interpretar0 el proceso o las ecuaciones para la solución. Pero tiene otra característica, y es que puede ser usado como entrada o salida de datos.

El dibujo no esta realizado en EES, pero se puede hacer en cualquier programa de dibujos como el Paint, Microsoft Draw, Corel Drawm, o Power Point. También puede ser una imagen escaneada. Se copia (Copy) el dibujo y se pega (Paste) dentro de la ventana **Diagram Window**.



Picando en **Show Diagram Tool Bar**, en el menú **Options**, aparece una barra de herramientas con posibilidades de añadir texto, líneas, figuras, y utilizar el **Diagram Window** como entrada y salida de datos.

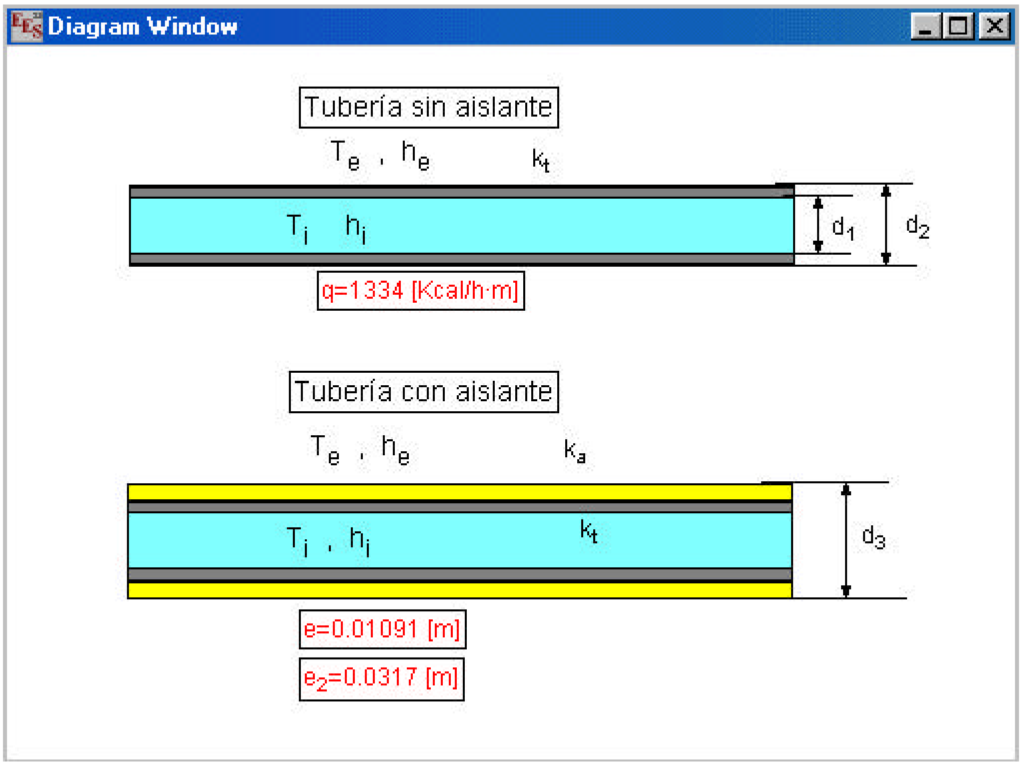


Una vez presentadas las herramientas más usuales y utilizadas de EES pasaremos a los problemas, que serán expuestos casi como sale en pantalla para que sirva de guía a la hora de iniciarse en la resolución de problemas con el programa EES.

3. Conducción en régimen permanente

3.1. Problema 1

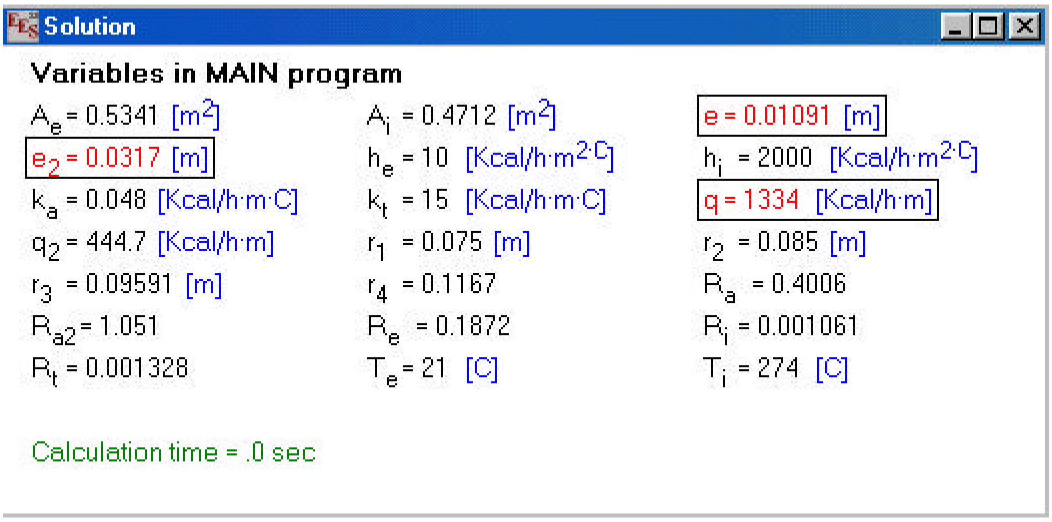
Por el interior de una tubería de acero, de 17 cm de diámetro exterior y de 15 cm de diámetro interior (conductividad térmica 15 Kcal/h m C), circula vapor saturado a 60 Kgf/cm^2 de presión (T=274 C) atravesando un local que se encuentra a 21 C.
Calcular:
1.- Flujo de calor por unidad de longitud.
2.- Espesor de aislante (lana de roca de conductividad térmica 0.048 Kcal/h m C) necesario para reducir el flujo de calor a la tercera parte.
3.- Espesor de aislante necesario para reducir la temperatura exterior hasta un valor máximo de 50 C.
Loas coeficientes de películas exterior e interior son 10 Kcal/2000 Kcal/m^{2}C respectivamente.



Equations Window

```
"Datos"
r_1=0.075 ;   r_2=0.085
T_i=274    ;   T_e=21
h_i=2000  ;    h_e=10
k_t=15       ;    k_a=0.048
"Solución"
"1.- Flujo de calor por unidad de longitud"
{Utilizando el concepto de resistencia térmica , se puede expresar el flujo de calor mediante:
Q=(T_i-T_e)/(suma de resistencias)
Refiriendo el cálculo a 1 m de tubería , tenemos}
A_i=2*pi*r_1   ;   A_e=2*pi*r_2
R_i=1/(h_i*A_i)                          {Resistencia por convección interior}
R_e=1/(h_e*A_e)                          {Resistencia por convección exterior}
R_t=ln(r_2/r_1)/(2*pi*k_t)         {Resistencia por conducción en la tubería}
q=(T_i-T_e)/(R_i+R_t+R_e)             { ecuación de la transmisión de calor  por conducción en un cilindro por unidad de longitud}
"2.- Espesor de aislante necesario para reducir la temperatura exterior hasta un valor máximo de 50 C."
{Para reducir el flujo de calor a la tercera parte, se añade aislante a la tubería , lo cual introduce una nueva resistencia térmica}
R_a=(ln(r_3/r_2))/(2*pi*k_a)       {Resistencia térmica del aislante }
q_2=q/3                             {reducimos el flujo de calor a la tercera parte}
q_2=(T_i-T_e)/(R_i+R_t+R_a+(1/(h_e*2*pi*r_3)))               { ecuación de la transmisión de calor  por conducción en un cilindro por unidad de longitud}
e=r_3-r_2                       {espesor de aislante necesario}
"3.- Espesor de aislante necesario para reducir la temperatura exterior hasta un valor máximo de 50 C."
{Vamos a tener otro espesor de aislante diferente que lo vamos a llamar e_2, y un nuevo radio, r_4}
R_a2=(ln(r_4/r_2))/(2*pi*k_a)          {Resistencia térmica del aislante del tercer apartado}
2*pi*r_4*h_e*(50-T_e)=(T_i-T_e)/(R_i+R_t+R_a2+(1/(h_e*2*pi*r_4)))
e_2=r_4-r_2
```

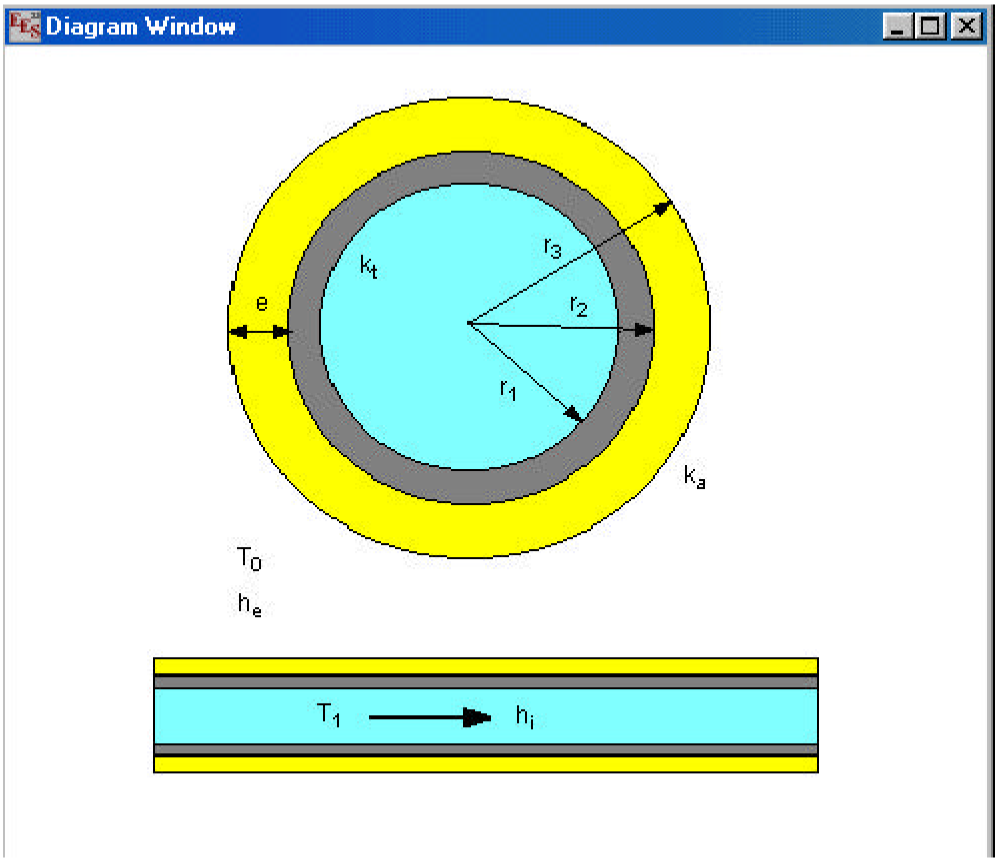
Una vez que tenemos escritas todas las ecuaciones y datos necesarios para solucionar el problema, picamos en **Solve**, en el menú **Calculate**, y obtenemos la solución.



Podemos ver las ecuaciones expresadas en forma matemática picando en **Formatted Equations**, en el menú **Windows.**

3.2. Problema 2

Un fluido a temperatura T_1 circula a través de una tubería cuyos radios interior y exterior miden r_1 y r_2 respectivamente. Por su parte externa, la tubería está en contacto con un fluido a temperatura constante T_0 menor que la temperatura del fluido. Para reducir la pérdida de calor desde el fluido interior se aisla la tubería. Estudiar la variación de la pérdida de calor de la tubería en función del espesor de aislamiento.
El coeficiente de película interior es h_i, el coeficiente de película exterior es h_e, la conductividad de la tubería es k_t y la conductividad del aislante k_a.



Aquí tenemos un esquema del problema, vamos a pasar a la ventana de ecuaciones.

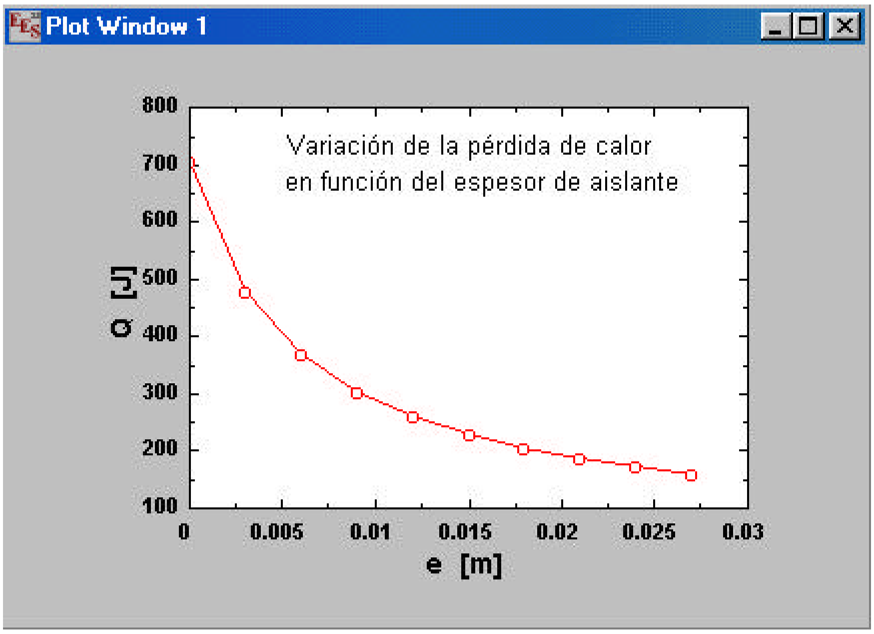
"Datos"
r_1=.025 { radio interior }
r_2=0.035 { radio exterior }
T_1=300 {Temperatura del fluido interior a la tubería}
T_0=20 {Temperatura del fluido exterior a la tubería}
h_i=2330 {coeficiente de película interior}
h_e=11.63 {coeficiente de película exterior}
k_t=17.5 {conductividad térmica de la tubería}
k_a=0.06 {conductividad térmica del aislante}
L=1 {longitud de la tubería}

"Solución"
A_i=2*pi*r_1
A_e=2*pi*r_3
R_i=1/(h_i*A_i) {Resistencia por convección interior}
R_e=1/(h_e*A_e) {Resistencia por convección exterior}
R_a=ln(r_3/r_2)/(2*pi*k_a) {Resistencia térmica del aislante }
R_t=ln(r_2/r_1)/(2*pi*k_t) {Resistencia por conducción en la tubería}
Q/L=(T_1-T_0)/(R_i+R_t+R_a+R_e) { ecuación de la transmisión de calor por conducción en un cilindro por unidad de longitud}
e=r_3-r_2 { espesor de aislante en función del radio r_2 y r_3}

Vamos a presentar la tabla paramétrica y los gráficos.

Parametric Table

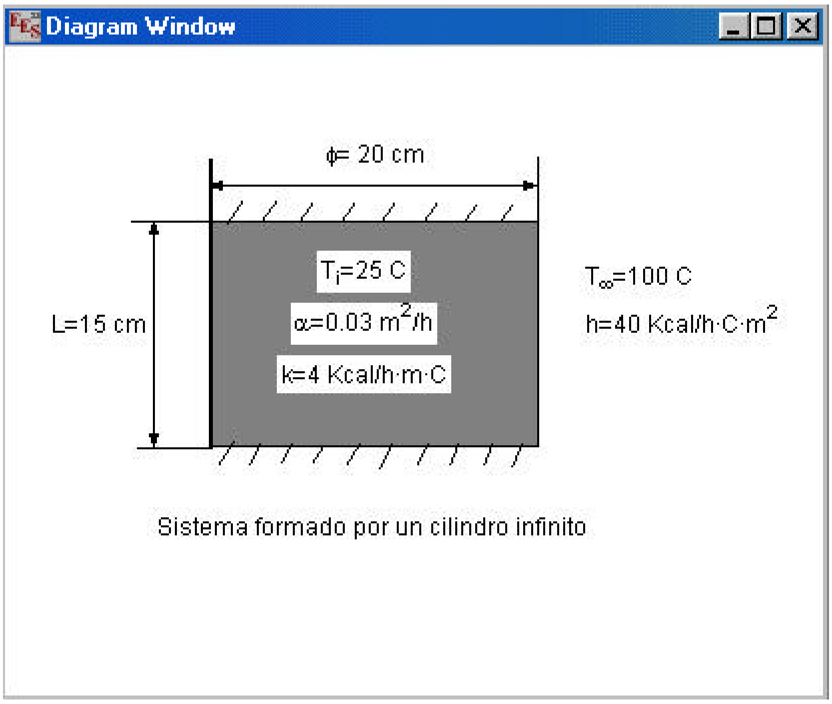
1..10	1 r_3 [m]	2 e [m]	3 Q [J]
Run 1	0.035	0	705.7
Run 2	0.038	0.003	479.4
Run 3	0.041	0.006	368.8
Run 4	0.044	0.009	303.1
Run 5	0.047	0.012	259.5
Run 6	0.05	0.015	228.5
Run 7	0.053	0.018	205.2
Run 8	0.056	0.021	187.1
Run 9	0.059	0.024	172.5
Run 10	0.062	0.027	160.6



4. Conducción en régimen transitorio

4.1. Problema 3

Un cilindro de 20 cm de diámetro y 15 cm de longitud, de un material de conductividad térmica 4 Kcal/h·m·C y difusividad térmica 0.03m^2/h, tiene tapados sus extremos lo que esencialmente lo aisla del flujo de calor. La temperatura del cilindro es uniforme, 25 C, y se pone en contacto con un fluido a 100 C.
Calcular la temperatura en el centro geométrico del cilindro en el plano central y en un punto situado a 5 cm del centro, una vez transcurridos 30 minutos desde el instante inicial.

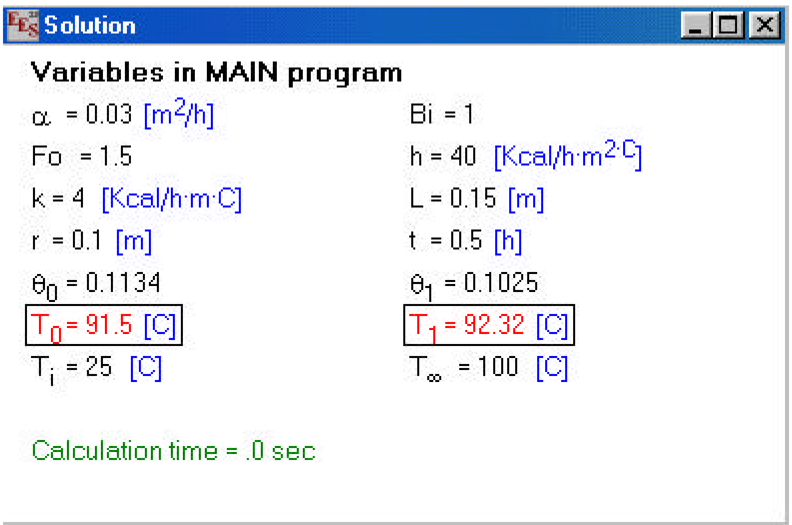


"Datos"
r=0.10
L=0.15
alpha=0.03; t=0.5
k=4
h=40
T_i=25
T_infinity=100
"Solución"
{Si el cilindro está aislado en sus dos extremos, el flujo de calor no puede tener componente axial en ningún punto. Así, es como si el cilindro fuese infinito. Vamos a necesitar el module heislercilindro}

```
MODULE heislercilindro(Bi,Fo,X,theta_c)
y*Bessel_J1(y)=Bi*Bessel_J0(y)
A_n=(2*Bessel_J1(y))/(y*((Bessel_J0(y))^2+(Bessel_J1(y))^2))
f_n=Bessel_J0(y*X)
theta_c=A_n*exp(-y^2*Fo)*f_n
END
Bi=(h*r)/k                                {número de Biot}
Fo=(alpha*t)/r^2                          {número de Fourier}
CALL heislercilindro(Bi,Fo,0,theta_0)     {Module que nos da la temperatura adimensional}
theta_0=(T_0-T_infinity)/(T_i-T_infinity)     { temperatura adimensional en el centro}
CALL heislercilindro(Bi,Fo,0.5,theta_1)   {Module que nos da la temperatura adimensional a un punto a 5 cm del centro}
theta_1=(T_1-T_infinity)/(T_i-T_infinity)     { temperatura adimensional en un punto a 5 cm del centro}
```

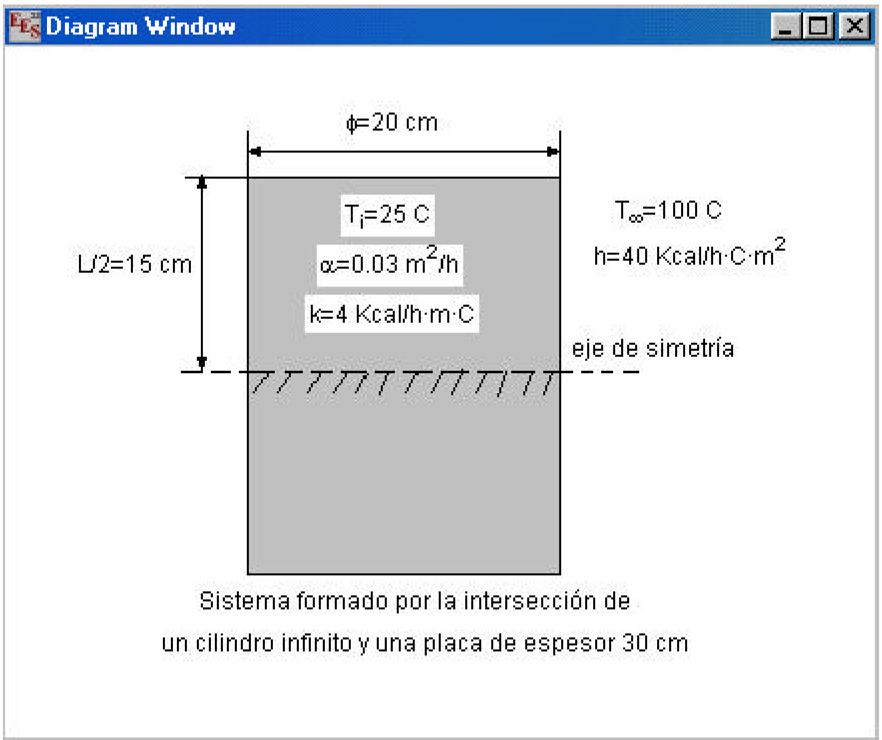
El MODULE heislercilindro(Bi,Fo,X,theta_c) ha sido introducido con el comando **Merge**, en el menú **File**.

Picando en **Solve**, en el menú **Calculate**, obtenemos la solución.



4.2. Problema 4

Un cilindro de 20 cm de diámetro y 15 cm de longitud, de un material de conductividad térmica 4 Kcal/h·m·C y difusividad térmica 0.03m^2/h, tiene tapados sus extremos lo que esencialmente lo aisla del flujo de calor. La temperatura del cilindro es uniforme, 25 C, y se pone en contacto con un fluido a 100 C.
Calcular la temperatura en el centro geométrico del cilindro en el plano central y en un punto situado a 5 cm del centro, una vez transcurridos 30 minutos desde el instante inicial, suponiendo que el cilindro tiene descubierto uno de sus extremos.



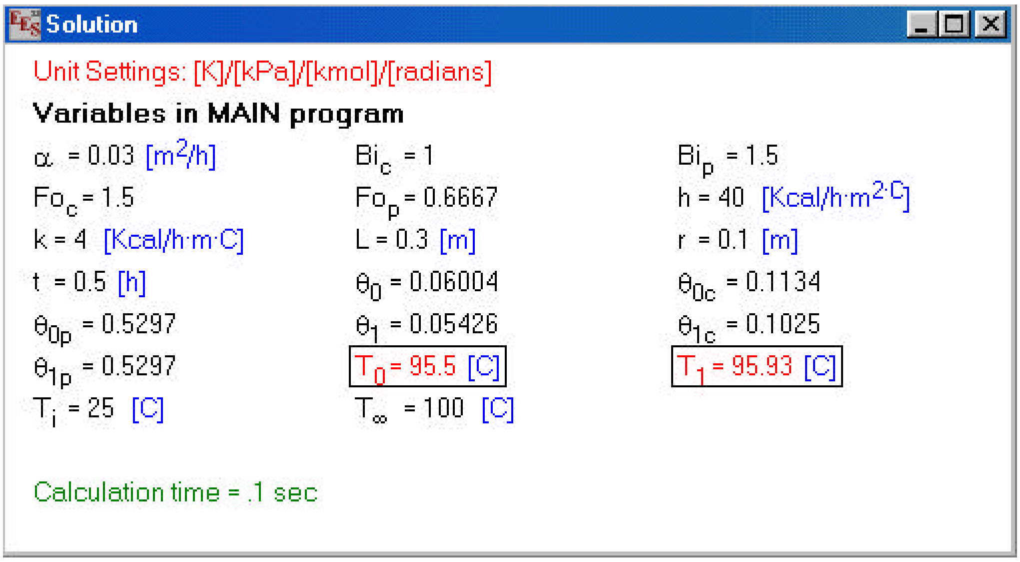
Equations Window

"Solución"
{El extremo aislado del cilindro puede asimilarse a una condición de simetría.Por ello el sistema puede considerarse formado por la intersección de un cilindro infinito y una placa de espesor doble de la altura del cilindro (ver diagram window).
Vamos a necesitar los modules, heislerplaca y heislercilindro}

```
MODULE heislerplaca(Bi,Fo,X,theta_p)
Bi*cos(y)=y*sin(y)
A_n=(2*sin(y))/(y+sin(y)*cos(y))
f_n=cos(y*X)
theta_p=A_n*exp(-y^2*Fo)*f_n
END
MODULE heislercilindro(Bi,Fo,X,theta_c)
y*Bessel_J1(y)=Bi*Bessel_J0(y)
A_n=(2*Bessel_J1(y))/(y*((Bessel_J0(y))^2+(Bessel_J1(y))^2))
f_n=Bessel_J0(y*X)
theta_c=A_n*exp(-y^2*Fo)*f_n
```

```
END
"Datos"
r=0.10
L=0.30
alpha=0.03;  t=0.5
k=4
h=40
T_i=25
T_infinity=100
Bi_c=(h*r)/k ;   Bi_p=(h*(L/2))/k          {número de Biot del cilindro y la placa respectivamente}
Fo_c=(alpha*t)/r^2  ;   Fo_p=(alpha*t)/((L/2)^2)   {número de Fourier del cilindro y la placa respectivamente}
theta_0=theta_0c*theta_0p
CALL heislercilindro(Bi_c,Fo_c,0,theta_0c)      {Module que nos da la temperatura adimensional}
CALL heislerplaca(Bi_p,Fo_p,0.5,theta_0p)      {El punto no está en el centro de la placa, sino a 7.5 cm del centro, por eso ponemos  X=x/L=0.5}
theta_0=(T_0-T_infinity)/(T_i-T_infinity)           { temperatura adimensional en el centro}
{Para calcular la temperatura en el punto a 5 cm del centro del cilindro, tenemos}
CALL heislercilindro(Bi_c,Fo_c,0.5,theta_1c)  {ahora X=0.5}
CALL heislerplaca(Bi_p,Fo_p,0.5,theta_1p)       { éste es igual que en la primera parte del problema}
theta_1=theta_1c*theta_1p
theta_1=(T_1-T_infinity)/(T_i-T_infinity)          { temperatura adimensional en  un punto a 5 cm del centro}
```

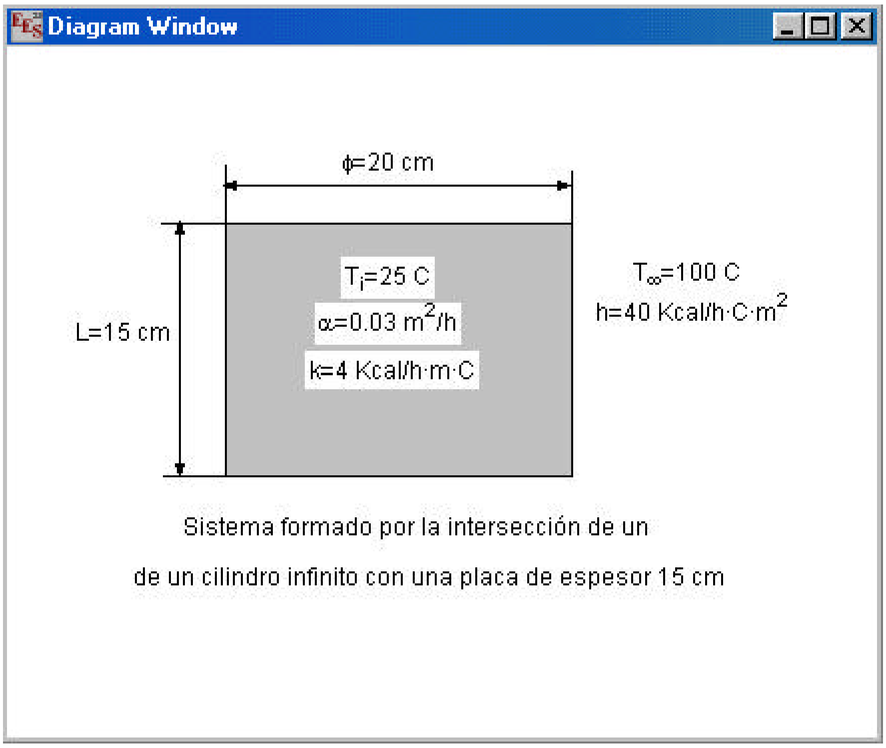
Picando en **Solve**, en el menú **Calculate**, obtenemos la solución.



4.3. Problema 5

Un cilindro de 20 cm de diámetro y 15 cm de longitud, de un material de conductividad térmica 4 Kcal/h·m·C y difusividad térmica 0.03m^2/h, tiene tapados sus extremos lo que esencialmente lo aisla del flujo de calor. La temperatura del cilindro es uniforme, 25 C, y se pone en contacto con un fluido a 100 C.

Calcular la temperatura en el centro geométrico del cilindro en el plano central y en un punto situado a 5 cm del centro, una vez transcurridos 30 minutos desde el instante inicial, suponiendo los dos extremos descubiertos. Calcular también la temperatura media del cilindro en este caso.



Equations Window

"Solución"
{En este caso, el sistema es la intersección de un cilindro infinito con una placa de espesor igual a la altura del cilindro, como se muestra en diagram window.
Vamos a necesitar los modules, heislercilindro, heislerplaca y para la temperatura adimensional, grobercilindro y groberplaca}

```
MODULE heislercilindro(Bi,Fo,X,theta_c)
y*Bessel_J1(y)=Bi*Bessel_J0(y)
A_n=(2*Bessel_J1(y))/(y*((Bessel_J0(y))^2+(Bessel_J1(y))^2))
f_n=Bessel_J0(y*X)
theta_c=A_n*exp(-y^2*Fo)*f_n
END

MODULE heislerplaca(Bi,Fo,X,theta_p)
Bi*cos(y)=y*sin(y)
A_n=(2*sin(y))/(y+sin(y)*cos(y))
f_n=cos(y*X)
```

```
theta_p=A_n*exp(-y^2*Fo)*f_n
END

MODULE grobercilindro(Bi,Fo,Q#Q_0c)
y*Bessel_J1(y)=Bi*Bessel_J0(y)
A_n=(2*Bessel_J1(y))/(y*((Bessel_J0(y))^2+(Bessel_J1(y))^2))
B_n=(2*Bessel_J1(y))/y
Q#Q_0c=1-(A_n*exp(-y^2*Fo)*B_n)
END

MODULE groberplaca(Bi,Fo,Q#Q_0p)
Bi*cos(y)=y*sin(y)
A_n=(2*sin(y))/(y+sin(y)*cos(y))
B_n=sin(y)/y
Q#Q_0p=1-(A_n*exp(-y^2*Fo)*B_n)
END

"Datos"
r=0.10
L=0.15
alpha=0.03;  t=0.5
k=4
h=40
T_i=25
T_infinity=100
Bi_c=(h*r)/k ;   Bi_p=(h*(L/2))/k          {número de Biot del cilindro y la placa respectivamente}
Fo_c=(alpha*t)/r^2  ;  Fo_p=(alpha*t)/((L/2)^2)   {número de Fourier del cilindro y la placa respectivamente}
theta_0=theta_0c*theta_0p
CALL heislercilindro(Bi_c,Fo_c,0,theta_0c)      {Module que nos da la temperatura adimensional}
CALL heislerplaca(Bi_p,Fo_p,0,theta_0p)      {X=0 ,porque está en el centro de la placa}
theta_0=(T_0-T_infinity)/(T_i-T_infinity)           { temperatura adimensional en el centro}
{Para calcular la temperatura en el punto a 5 cm del centro del cilindro, tenemos}
CALL heislercilindro(Bi_c,Fo_c,0.5,theta_1c)  {ahora X=0.5}
CALL heislerplaca(Bi_p,Fo_p,0,theta_1p)      { X=0, porque está en el centro de la placa}
theta_1=theta_1c*theta_1p
theta_1=(T_1-T_infinity)/(T_i-T_infinity)          { temperatura adimensional en un punto a 5 cm del centro}
{Vamos a calcular la temperatura media, theta_m=1-(Q/Q_i)}
theta_m=theta_mc*theta_mp          {temperatura media, es el producto de la temperatura media del cilindro y de la placa}
theta_mc=1-Q#Q_0c
theta_mp=1-Q#Q_0p
CALL  grobercilindro(Bi_c,Fo_c,Q#Q_0c)      {module que nos proporcional el porcentaje de calor transferido sobre el total posible, es el ábaco de Grober}
```

CALL groberplaca(Bi_p,Fo_p,Q#Q_0p)
theta_m=(T_m-T_infinity)/(T_i-T_infinity)

Picando en **Solve**, en el menú **Calculate**, obtenemos la solución.

Solution

Unit Settings: [K]/[kPa]/[kmol]/[radians]

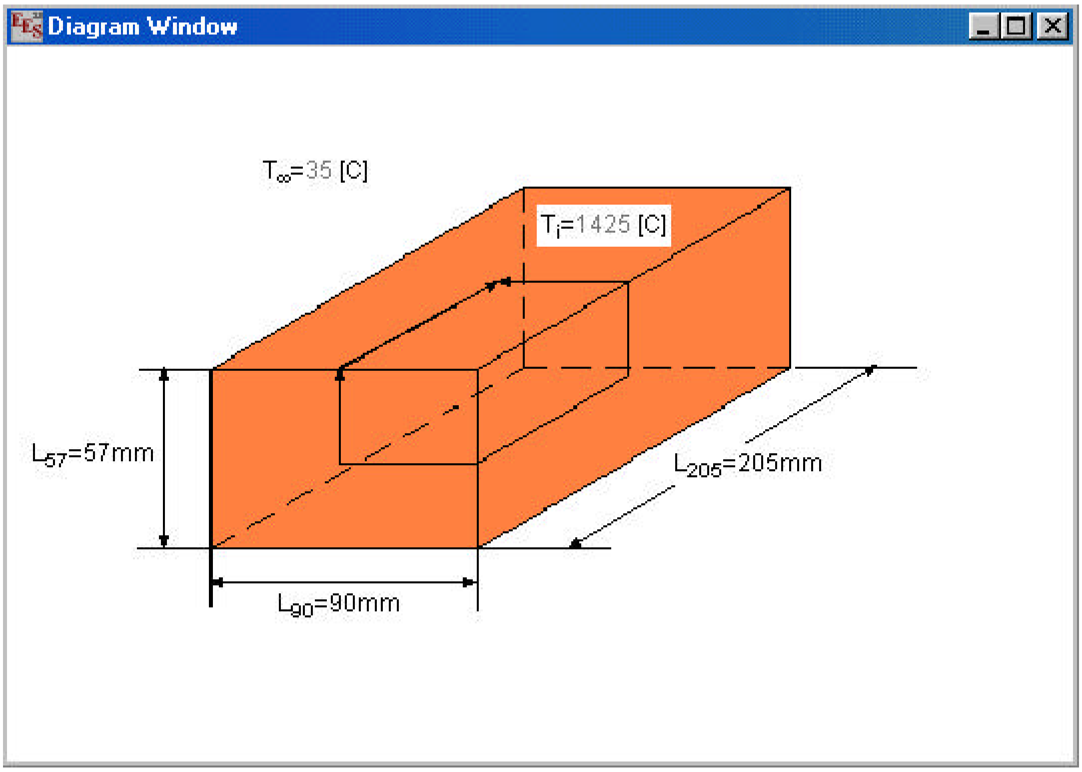
Variables in MAIN program

α = 0.03 [m^2/h]	Bi_c = 1	Bi_p = 0.75
Fo_c = 1.5	Fo_p = 2.667	h = 40 [Kcal/h·m^2·C]
k = 4 [Kcal/h·m·C]	L = 0.15 [m]	$Q\#Q_{0c}$ = 0.9076
$Q\#Q_{0p}$ = 0.7972	r = 0.1 [m]	t = 0.5 [h]
θ_0 = 0.02544	θ_{0c} = 0.1134	θ_{0p} = 0.2244
θ_1 = 0.02299	θ_{1c} = 0.1025	θ_{1p} = 0.2244
θ_m = 0.01875	θ_{mc} = 0.09243	θ_{mp} = 0.2028
$\boxed{T_0 = 98.09\ [C]}$	$\boxed{T_1 = 98.28\ [C]}$	T_i = 25 [C]
T_∞ = 100 [C]	$\boxed{T_m = 98.59\ [C]}$	

Calculation time = .1 sec

4.4. *Problema 6*

Un ladrillo de 57x90x205 mm se cuece en un horno a 1425 C.Se enfría en aire a 35 C.
a) Calcular el tiempo para que la temperatura en el centro de la cara de 90x205 sea 70C. El coeficiente de transferencia convectivo entre el ladrillo y el aire es 20 Kcal/h·m^2·C. El material del ladrillo tiene un calor específico de 0.2 Kcal/KgC, una densidad de 1600 Kg/m^3 y una conductividad térmica de 0.6 Kcal/h·m·C.
b) Calcular el tiempo para que la temperatura en el centro de la cara de 90x205 sea 70 C, haciendo una comparación de la variación del tiempo en función de la conductividad térmica.



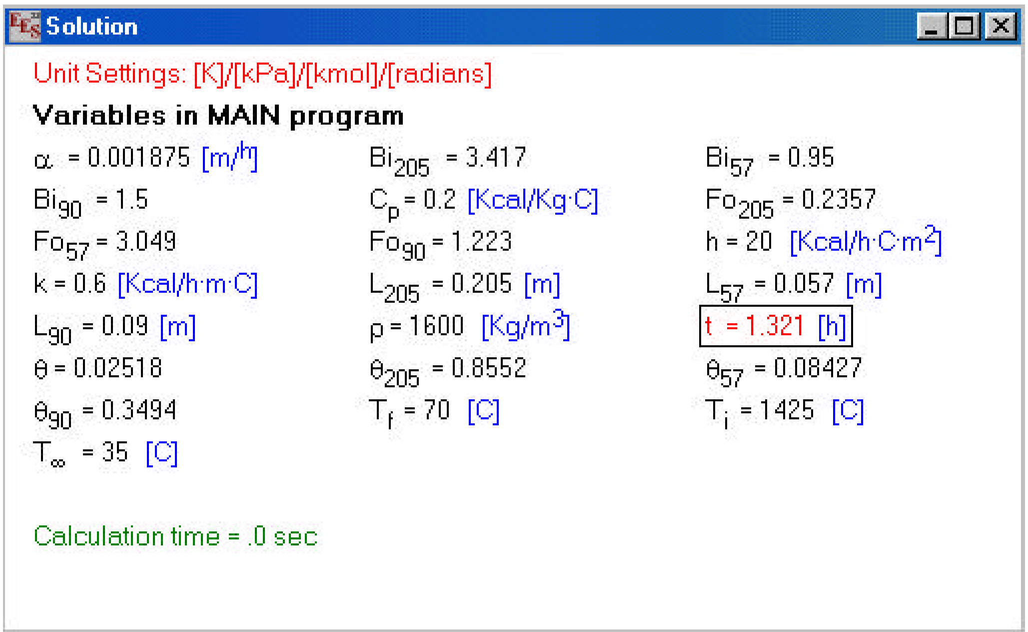
a)

Equations Window

```
"Solución"
{El ladrillo es la intersección de tres placas perpendiculares entre sí de dimensiones
57, 90 y 205 mm, como se ve en diagram window.El punto que nos dicen está en la
superficie de la placa de 57 mm de espesor y en el centro de las otras dos. La
temperatura adimensional de este punto será:
theta=theta_57(X=1)*theta(X=0)*theta(X=0)
Vamos a necesitar el module heislerplaca}
MODULE heislerplaca(Bi,Fo,X,theta_p)
Bi*cos(y)=y*sin(y)
A_n=(2*sin(y))/(y+sin(y)*cos(y))
f_n=cos(y*X)
theta_p=A_n*exp(-y^2*Fo)*f_n
```

```
END
"Datos"
L_57=0.057;   L_90=0.09;   L_205=0.205
C_p=0.2
rho=1600
k=0.6
h=20
T_i=1425;  T_infinity=35;  T_f=70
"Ecuaciones para la solución"
alpha=k/(rho*C_p)
theta=(T_f-T_infinity)/(T_i-T_infinity)                {temperatura adimensional}
Bi_57=h*(L_57/2)/k;  Fo_57=(alpha*t)/(L_57/2)^2
Bi_90=h*(L_90/2)/k;  Fo_90=(alpha*t)/(L_90/2)^2
Bi_205=h*(L_205/2)/k;  Fo_205=(alpha*t)/(L_205/2)^2
CALL heislerplaca(Bi_57,Fo_57,1,theta_57)           {X=1, está en la superficie}
CALL heislerplaca(Bi_90,Fo_90,0,theta_90)           {X=0, está en el centro}
CALL heislerplaca(Bi_205,Fo_205,0,theta_205)      {X=0, está en el centro}
theta=theta_57*theta_90*theta_205                  {temperatura adimensional es el
producto de las tres temperaturas adimensionales}
```

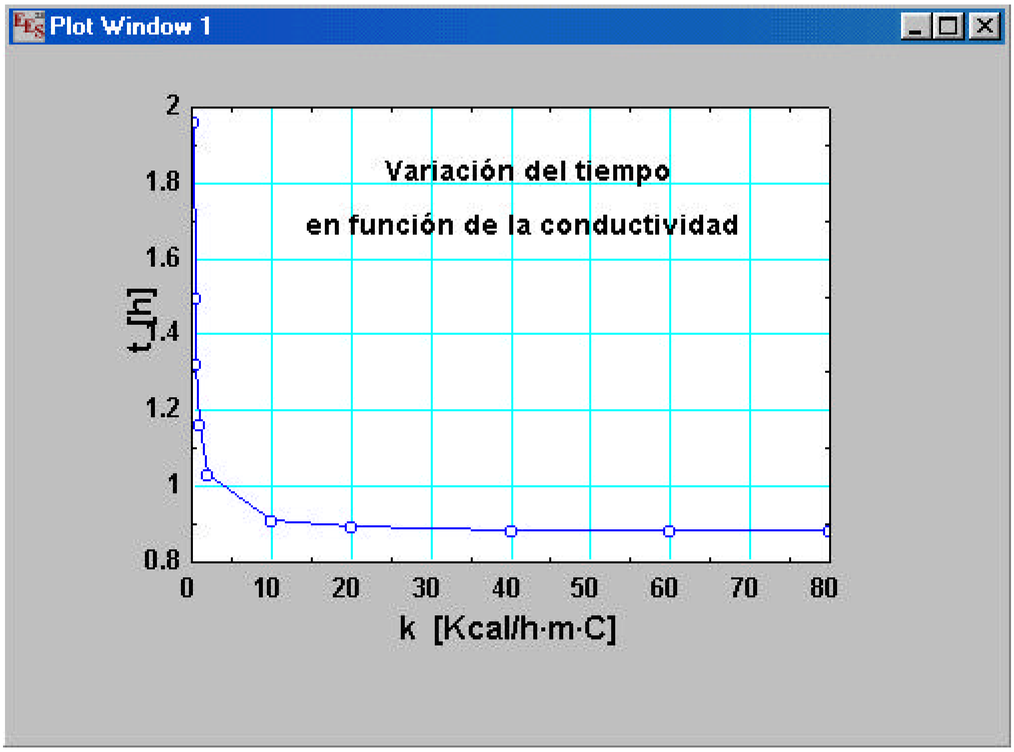
Picando en **Solve**, en el menú **Calculate**, obtenemos la solución.



b) Para este apartado Equations Window es el mismo pero encerrando entre paréntesis la conductividad térmica, ya que vamos a hacer un análisis paramétrico en función de esta variable. Vamos a hacer también la representación gráfica de los valores obtenidos.

Parametric Table

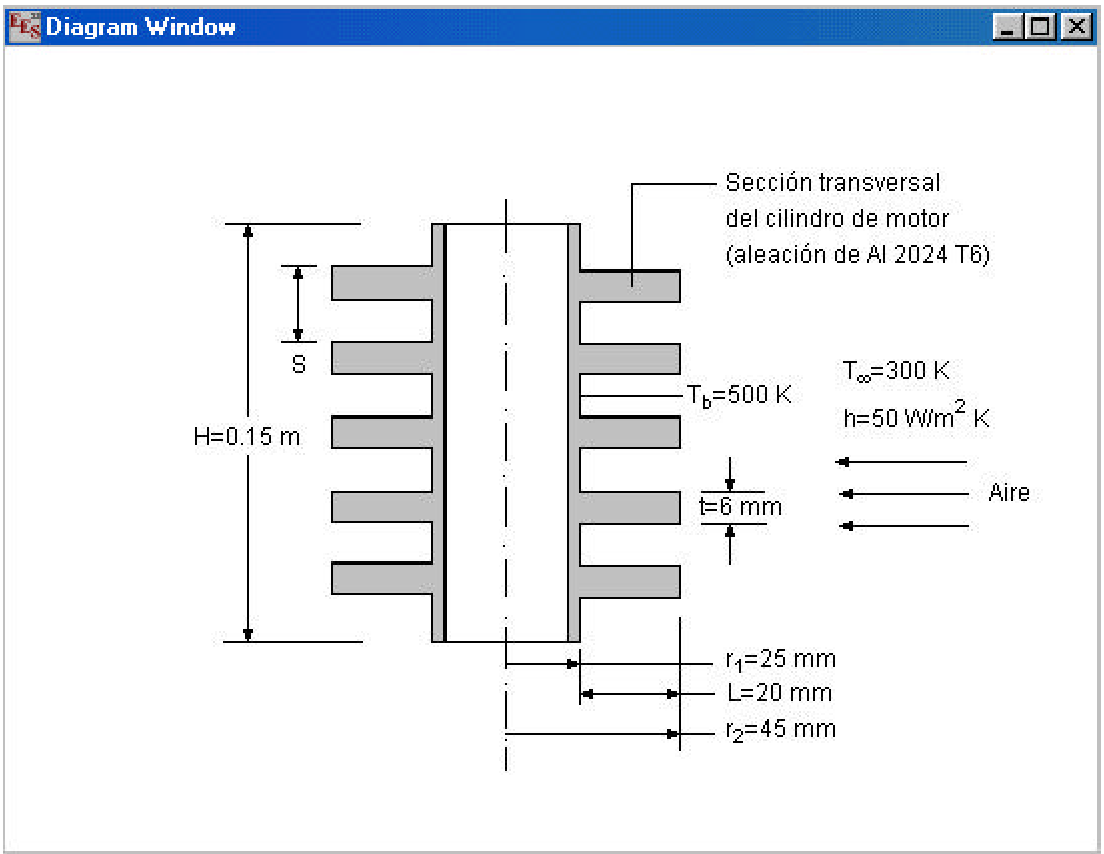
1..10	k [Kcal/h·m·C]	t [h]
Run 1	0.2	1.965
Run 2	0.4	1.5
Run 3	0.6	1.321
Run 4	1	1.163
Run 5	2	1.031
Run 6	10	0.9111
Run 7	20	0.8949
Run 8	40	0.8867
Run 9	60	0.8839
Run 10	80	0.8825



5. Aletas

5.1. Problema 7

Se conocen las condiciones de operación de un cilindro de motocicleta con aletas (ver 'diagram window'). Encontrar el aumento en la transferencia de calor asociada con el uso de aletas.

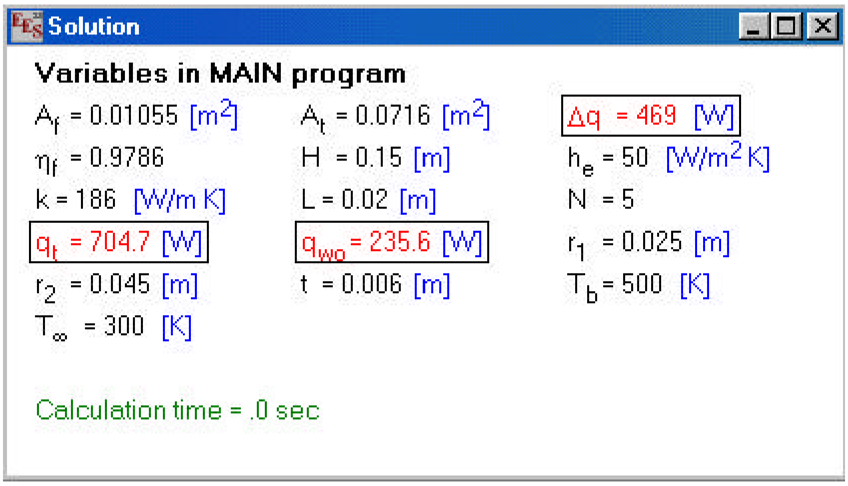


Equations Window

"Solución"
{Suposiciones:
1.- Condiciones de estado estable.
2.- Conducción radial unidimensional en las aletas.
3.- Propiedades constantes.
4.- Intercambio de radiación insignificante con los alrededores.
5.- Coeficiente de convección uniforme sobre la superficie externa (con o sin aletas)}
"Datos"
H=0.15; r_1=0.025; L=0.020; r_2=0.045; t=0.006 {Datos de la geometría}
T_b=500; h_e=50; T_infinity=300 {Datos de las condiciones}
k=186 {conductividad del aluminio en torno a T=400 K}
"Calor transferido con el uso de aletas"

q_t=h_e*A_t*(1-((N*A_f/A_t)*(1-eta_f)))*(T_b-T_infinity)
A_t=N*A_f+(2*pi*r_1*(H-(N*t))) {área total}
A_f=Af_aletcircrectang(r_1,r_2,L,t) {área de la aleta obtenida de función de librería}
N=5 {número de aletas}
eta_f=Ef_aletcircrectang(h_e,k,r_1,r_2,t) {eficiencia de la aleta, función de librería}
"calor transferido sin aletas"
q_wo=h_e*(2*pi*r_1*H)*(T_b-T_infinity)
"Diferencia de calor transferido entre un tipo y otro"
DELTAq=q_t-q_wo

Picando en **Solve**, en el menú **Calculate**, obtenemos la solución.

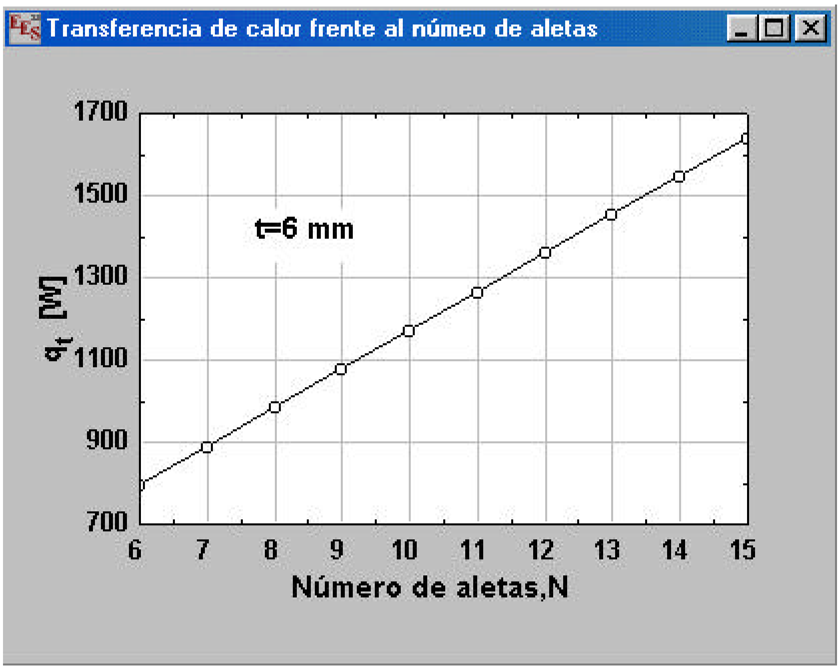


Vamos a ver cómo varía el calor transferido en función del número de aletas. En otro fichero de EES resolvemos el análisis paramétrico.

{Aunque las aletas aumentan de manera significativa la transferencia de calor del cilindro, aún es posible un mejoramiento considerable si se aumenta el número de aletas. Evaluamos esta posibilidad viendo la evolucin de q_t en función de N manteniendo primero fijo el espesor de la aleta (t=6 mm) e incrementando el número de aletas al reducir el espacio entre las aletas.Determinamos un espaciado de aletas de 2 mm en cada extremo del arreglo y un hueco mínimo de aleta de 4 mm.}
N_max=H/S {número máximo de aletas}
S=0.002*2+0.006
q_t=h_e*A_t*(1-((N*A_f/A_t)*(1-eta_f)))*(T_b-T_infinity)
A_f=Af_aletcircrectang(r_1,r_2,L,t) {área de la aleta obtenida de función de librería}
A_t=N*A_f+(2*pi*r_1*(H-(N*t))) {área total en caso b}
eta_f=Ef_aletcircrectang(h_e,k,r_1,r_2,t) {eficiencia de la aleta, función de librería}

Parametric Ta...

1..10	N	q_t [W]
Run 1	6	798.5
Run 2	7	892.3
Run 3	8	986.1
Run 4	9	1080
Run 5	10	1174
Run 6	11	1267
Run 7	12	1361
Run 8	13	1455
Run 9	14	1549
Run 10	15	1643



{El número de aletas también aumenta reduciendo el espesor de la aleta. Si el hueco de la aleta se fija en (S-t)=4 mm y las restricciones de fabricación dictan un espesor de aleta mínimo permisible de 2 mm, se pueden acomodar hasta N=25 aletas, en este caso el estudio paramétrico queda}

t=0.002 {espesor de aleta mínimo permisible}

S-t=0.004

N_max=H/S

q_t=h_e*A_t*(1-((N*A_f/A_t)*(1-eta_f)))*(T_b-T_infinity)

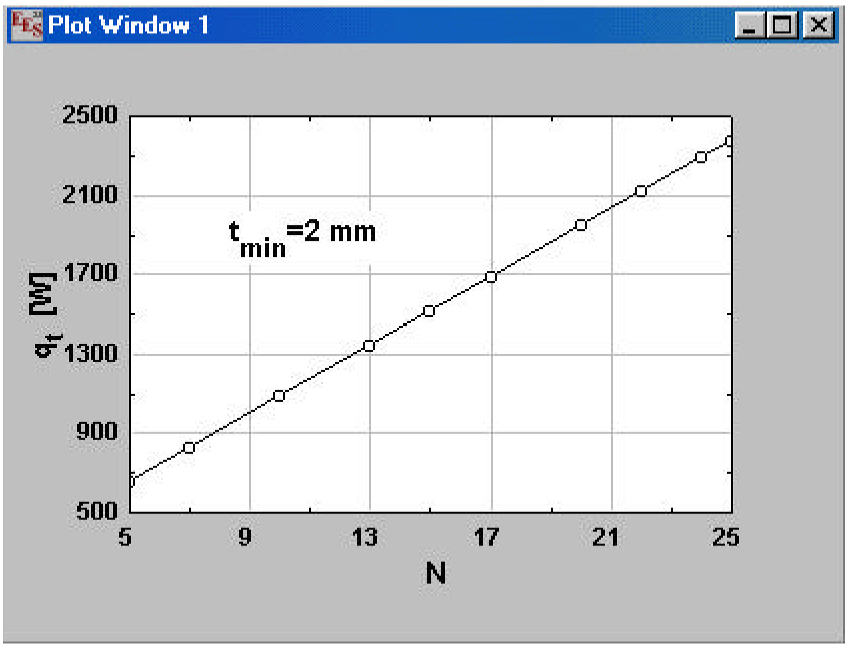
A_t=N*A_f+(2*pi*r_1*(H-(N*t))) {área total }

A_f=Af_aletcircrectang(r_1,r_2,L,t) {área de la aleta obtenida de función de librería}

eta_f=Ef_aletcircrectang(h_e,k,r_1,r_2,t) {eficiencia de la aleta, función de librería}

Parametric Table

1..10	N	q_t [W]
Run 1	5	664.6
Run 2	7	836.2
Run 3	10	1094
Run 4	13	1351
Run 5	15	1522
Run 6	17	1694
Run 7	20	1951
Run 8	22	2123
Run 9	24	2295
Run 10	25	2380



6. Convección

6.1. Problema 8

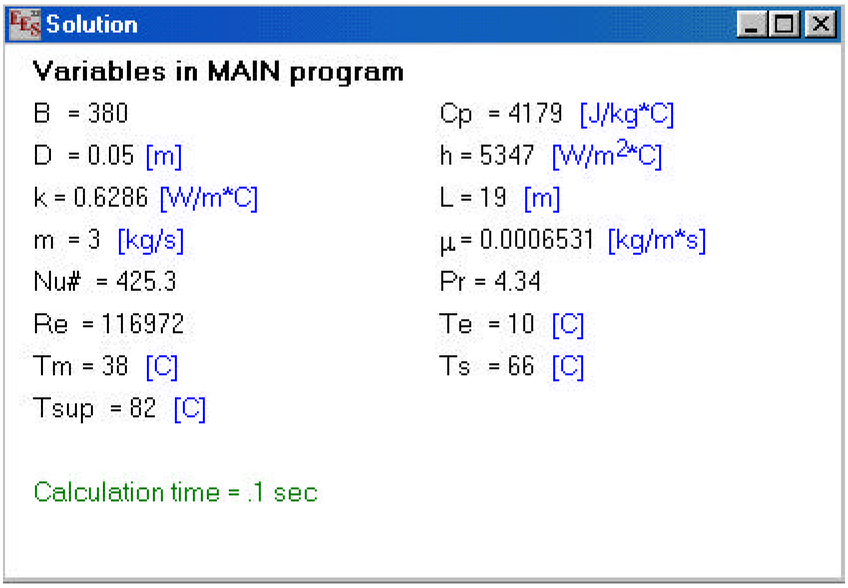
Se desean calentar 3 kg/s de agua que circula a través de una tubería horizontal de 5 cm de diámetro, desde 10 C hasta 66 C. La temperatura de la superficie de la tubería se mantiene a 82 C. Determinar:

a) Longitud de tubería necesaria para alcanzar la temperatura del agua a la salida.
b) coeficiente local de transferencia de calor en la superficie.

Equations Window

```
"Datos del problema"
Te=10                    { Temperatura de entrada}
Ts=66                    { Temperatura de salida}
Tsup=82                  { Temperatura superficial de la tubería}
m=3                      { gasto}
D=0.05                   {diámetro de la tubería }
Tm=(Ts+Te)/2             { Temperatura de masa }
"propiedades a la temperatura de masa"
Cp=4179                  {Calor específico}
mu=0.0006531             {Viscosidad}
k=0.6286                 {Conductividad térmica }
Pr=4.34                  {Número de Prandtl}
Re=4*m/(pi*D*mu)         {Número de Reynolds}
B=L/D                    {Cociente entre la longitud de la tubería y su diámetro}
"Correlaciones y ecuaciones de balance"
Nu#=CFFI14(Re,Pr,B)      { Número de Nusselt mediante función de 'User library routines'}
Nu#=h*D/k                {Número de Nusselt en función del coeficiente de película }
m*Cp*(Ts-Te)=h*pi*D*L*(Tsup-Tm)   {Balance global de energía}
```

Picando en **Solve**, en el menú **Calculate**, obtenemos la solución.



Vamos a hacer un estudio paramétrico de la variación de la longitud necesaria cuando varía el gasto másico.

1..10	m [kg/s]	L [m]
Run 1	1	15.25
Run 2	2	17.52
Run 3	3	19
Run 4	4	20.12
Run 5	5	21.04
Run 6	6	21.82
Run 7	7	22.51
Run 8	8	23.12
Run 9	9	23.67
Run 10	10	24.17