Sección:__ Apellido:_____Nombre:_____Firma:_____
CC1001-Computación I – Control Nº 1: **Pregunta 1** – sábado 4 de abril de 2009 – Tpo total: 1 hr 45 minutos
Con apuntes individuales – Sin consultas - Contestar en esta hoja (NO use el reverso)

A)(2 ptos) Un entero positivo es perfecto si la suma de sus divisores es igual al mismo nº. Por ej, 6 es perfecto porque 1+2+3=6. Al respecto, escriba una función que reciba un número entero positivo y entregue true (verdadero) si es perfecto o false si no lo es.

```
static public boolean perfecto(int x){
  int suma=0;

  //repetir para i=1,2,…,x/2: 0.75 ptos (o la versión equivalente con while)
  for(int i=1; i<=x/2; ++i)

    //verificar si es divisor: 0.25 ptos
    if(x%i == 0)
      //acumular divisor: 0.5 ptos (incluye 0.1 por inicializar suma)
      suma = suma + i;

  //devolver valor verdadero o falso: 0.5 ptos
  return suma == x;//if(suma==x) return true; else return false;
}
```

B)(4 ptos) Escriba un programa que use la función anterior para determinar y escribir el número perfecto más cercano a 1000.

```
//determinar perfecto más cercano a 1000 por defecto: 1.5 ptos
int menor=999;          //0.25
while(!perfecto(menor)) //1.0
  --menor;              //0.25

//determinar perfecto más cercano a 1000 por exceso: 1.5 ptos
int mayor=1001;         //0.25
while(!perfecto(mayor)) //1.0
  ++mayor;              //0.25

//determinar perfecto más cercano: 1.0 ptos
if(mayor-1000 < 1000-menor)  //0.5
  U.println(mayor);          //0.2
else                         //0.1
  U.println(menor);          //0.2
```

Sección:__ Apellido:_____Nombre:_____Firma:_____
CC1001-Computación I – Control Nº 1: **Pregunta 2** – sábado 4 de abril de 2009 – Tpo total: 1 hr 45 minutos
Con apuntes individuales – Sin consultas - Contestar en esta hoja (NO use el reverso)

Escriba un programa que le permita a un conductor conocer el rendimiento de su auto en kms/litro, ingresando una lista de valores de kms recorridos (nº real) y litros consumidos (nº real). El fin de la lista se indica por un valor de 0 kms. El programa debe establecer el diálogo que se muestra en el siguiente ejemplo:

kms? 180.8
litros? 16
rendimiento=11.25
kms? 250
litros? 20
rendimiento=12.5

…
kms? 0 (indicación de fin de los datos)
resultados finales:
rendimiento promedio=13.2
mejor rendimiento=16.4
2º mejor rendimiento=16.1

```
//inicialización de datos globales: 0.5 ptos
double suma=0, primero=0, segundo=0;
int n=0;
//leer mientras kms != 0: 1.0 ptos
while(true){
  int kms=U.readDouble("kms?");
  if(kms==0) break;
  //calculo y escritura de rendimiento: 0.5 ptos
  double litros=U.readDouble("litros?");
  double rend=kms/litros;
  U.println("rendimiento="+rend);
  //acumulación de rendimiento: 0.5 ptos
  suma=suma+rend;
  n=n+1;
  //mantener los dos mejores rendimientos: 3.0 ptos
  if(rend>primero){     //0.5 (se acepta rend>=primero)
    segundo=primero;    //0.5
    primero=rend;       //0.5
  }else if(rend>segundo)//1.0
    segundo=rend;       //0.5
}
//escritura de resultados finales: 0.5 ptos
U.println("rendimiento promedio="+suma/n);
U.println("mejor rendimiento="+primero);
U.println("2º mejor rendimiento="+segundo);
-------------------------------------------------
//alternativa 2: leer mientras kms != 0: 1.0 ptos
int kms=U.readDouble("kms?");
while(kms!=0){
  …
  kms=U.readDouble("kms?");
}
```