Auxiliar Semana 1

Pregunta 1

Cree una funcion con su receta de disenno que reciba un numero de dos digitos los invierta de posicion.

EG: invertir(24) -> 42

```
# invertir(n): int -> int
# Invierte un numero de dos cifras
# EG invertir(24) -> 42
def invertir(n):
  decena = n//10
  unidad = n%10
  return unidad*10+decena

assert invertir(24)==42
assert invertir(12)==21
```

Pregunta 2

Cree una funcion con su receta de disenno que reciba un entero N y retorne la suma de todos los enteros desde 1 hasta N.

EG: sumEnteros(100) = 5050

```
# sumEnteros(n): int -> int
# Retorna la suma de todos los enteros hasta N
# EG: sumEnteros(100) = 5050
def sumEnteros(n):
  return n*(n+1)//2

assert sumEnteros(100)==5050
assert sumEnteros(1)==1
```

Pregunta 3

Cree una funcion con su receta de disenno que reciba dos enteros X, Y ($X \leq Y$) y retorne la suma de todos los enteros desde X hasta Y.

EG: sumEnterosEnRango(5, 10) = 40

```
# sumEnterosEnRango(x, y): int int -> int
# Retorna la suma de todos los enteros desde x hasta y
# EG: sumEnterosEnRango(5,10) = 40
def sumEnterosEnRango(x, y):
  return sumEnteros(y)-sumEnteros(x-1)

assert sumEnterosEnRango(5, 10)==45
assert sumEnterosEnRango(1,100)==5050
```

Pregunta 4

Cree una funcion **hoyEs(n)** con su respectiva receta de disenno que reciba un entero de ocho digitos de la forma YYYYMMDD, y retorne un string que diga "Hoy es DD del MM del anno YYYY"

EG: hoyEs(20200831) -> "Hoy es 31 del 08 del anno 2020"

```
# hoyEs: int -> str
# retorna una frase con la fecha que se le entrega en formato YYYYMMDD
# EG: hoyEs(20200831) -> "Hoy es 31 del 8 del anno 2020"
def hoyEs(n):
  dia = str(n%100)
  anno = str(n//10000)
  mes = str((n//100)%100)
  frase = "Hoy es " + dia + " del " + mes + " del anno " + anno
  return frase

assert hoyEs(20200831)=="Hoy es 31 del 8 del anno 2020"
```

Pregunta 5

Cree una funcion **minToHora(min)** con su receta de disenno que reciba una cantidad entera de minutos y retorne un string de la forma "HH:MM" que represente la cantidad de minutos entregados en horas

EG: minToHora(150)->"2:30"

```
# minToHora: int -> str
# retorna una string con los minutos entregados en formato hora
# EG: minToHora(150) -> "2:30"
def minToHora(n):
  mins = str(n%60)
  horas = str(n//60)
  return horas+":"+mins

assert minToHora(150)=="2:30"
```

Pregunta 6

Usando las funciones hoyEs y minToHora cree una nueva funcion **ahoraEs(n):** que reciba un entero de 12 digitos de la forma YYYYMMDDHHHH y retorne la frase "Hoy es DD del MM del anno YYYY, y la hora actual es JJ:KK", donde JJ:KK son HHHH minutos transformados a horas.

EG: ahoraEs(202008310150) -> "Hoy es 31 del 8 del anno 2020, y la hora actual es 2:30"

```
# ahoraEs: int -> str
# retorna una string con la informacion de la fecha y hora actuales.
# EG: ahoraEs(202008310150) -> "Hoy es 31 del 8 del anno 2020, y la hora actual es 2:30"
def ahoraEs(n):
  fecha = hoyEs(n//10000)
  hora = minToHora(n%10000)
  return fecha+", y la hora actual es "+hora

assert ahoraEs(202008310150)=="Hoy es 31 del 8 del anno 2020, y la hora actual es 2:30"
```