

AeroDescuentos: Se está desarrollando para una aerolínea su módulo de liquidación de tiquetes aéreos. Para el mismo, se tiene una función que aplica descuentos a la tarifa base del vuelo dependiendo del tiempo de antelación de la reserva y la edad del pasajero. Los descuentos SON ACUMULABLES.

Normativa 005, sobre los descuentos:

15% de descuento sacando el billete con antelación superior a 20 días.

5% a los pasajeros con edad inferior a 18 años y 8% a los pasajeros con edad superior a 65 años.

La siguiente es la especificación de la función que se usará en el módulo del cálculo de los descuentos:

```
/**
 * calcular la tarifa de cada billete según el trayecto, la antelación en la que se obtiene el billete y
 * la edad del pasajero, de acuerdo con la normativa 005.
 * @param tarifaBase valor base del vuelo
 * @param diasAntelacion dias de antelación del vuelo @param edad - edad del pasajero
 * @throws ExcepcionParametrosInvalidos [XXXXXXXXXXXXXXXXXXXXXXXXXXXX]
 */
public long calculoTarifa(long tarifaBase, int diasAntelacion, int edad)
```

De acuerdo con lo indicado, y teniendo en cuenta que NO hay precondiciones, en qué casos se debería arrojar una excepción de tipo ExcepcionParametrosInvalidos?. Agregue esto a la especificación.

```
/**
 * calcular la tarifa de cada billete según el trayecto, la antelación en la que se obtiene el billete y
 * la edad del pasajero, de acuerdo con la normativa 005.
 * @param tarifaBase valor base del vuelo
 * @param diasAntelacion dias de antelación del vuelo @param edad - edad del pasajero
 * @throws ExcepcionParametrosInvalidos [XXXXXXXXXXXXXXXXXXXXXXXXXXXX]
 * Arroja excepcion cuando se le ingresan valores negativos (los días, la edad, la tarifa base) y
 * cuando se le ingresa una edad exagerada.
 */
```

En la siguiente tabla enumere un conjunto de clases de equivalencia que -según usted- creen una buena división del conjunto de datos de entrada de la función anterior:

Número	Clase de equivalencia (en lenguaje natural o matemático).	Resultado correcto / incorrecto.
1	Edades negativas	INCORRECTO
2	Edad fuera de lo común	INCORRECTO
3	Edad menor de la mayoría de edad [0,18)	CORRECTO
4	Edad de adulto mayor (65,120)	CORRECTO
5	Días negativos	INCORRECTO
6	Días mayores de 20	CORRECTO
7	Tarifa base negativa	INCORRECTO

Para cada clase de equivalencia, defina un caso de prueba específico, definiendo: parámetros de entrada y resultados esperados. A partir de las clases de equivalencia identificadas en el punto 2, identifique las condiciones límite o de frontera de estas. Para cada una de las condiciones de frontera anteriores, defina casos de prueba específicos.

Edades negativas, probar edades de -1, 0 y 1. Debe fallar el negativo y los otros dos pasar.
Edad de 119, 120 y 121, fuera de lo común. Debería solo pasar el 119.
Mayoría de edad, probar con 17, 18 y 19. Debe solo ser válido con 17 el descuento, con los otros dos no.
Adulto mayor, probar con 65, 66 y 67. Debería aplicarse el descuento en el caso de 66 y 67, en el 65 no.
Días negativos, probar días, -1, 0 y 1. Debe fallar el negativo y los otros dos pasar.
Días de descuento, probar con 20, 21 y 22. Debe aplicarse el descuento en 21 y 22, el 20 no debe.
Tarifa base negativa, probar tarifas de -1, 0 y 1. Debe fallar el negativo y los otros dos pasar.