Pregunta 3b

•••

"Una imagen vale mil palabras", pero consume mil veces más memoria.

Codigo Con Sustitucion de Constantes

```
class Abra {
  int a = 2, b = 1
  fun cus(int x): int {
    a = b + x
    return pide(a)
  }
  fun pide(int y): int {
    return a - y * b
  }
  class Cadabra extends Abra {
    Abra zo = new PataDeCabra()
    fun pide(int y): int {
       return zo.cus(a + b) - y
    }
}
```

```
class PataDeCabra extends Cadabra {
  int b = 7, c = 6
  fun cus(int x): int {
    a = x - 3
    c = a + b * c
    return pide(a * b + x)
  }
  fun pide(int y): int {
    return c - y * a
}
```

Se consideran las variables a,b, c y zo como Variables de instancia dado que no son statics (similar a java)

Asociación Dinámica de Métodos

Abra ho = new Cadabra()

Abra po = new PataDeCabra()

Cadabra cir = new PataDeCabra()

print(ho.cus(1) + po.cus(1) + cir.cus(1))

Inicialmente se crea una variable de tipo Cadabra y dos de tipo PataDeCabra.

Por emplearse asociación dinamica de métodos se tiene que cir y po tendrán los métodos de PataDeCabra. Y dado que Cadabra extiende a Abra y sobre-escribe la función pide entonces ho tiene dos métodos: cus (de Abra) y pide (de Cadabra).

Como el espacio para visualizar la ejecución es limitado se mostrará únicamente la línea print y se mostrará el método que se esté ejecutando.

```
class Abra {
    int a = 2, b = 1
0    fun cus(int x): int {
1         a = b + x
2         return pide(a)
     }
    fun pide(int y): int {
3         return a - y * b
     }
}
```

ho

Abra

Cad abra

cus

		po		cir			
		cus	PataDe Cabra	cus	PataDe Cabra		
	pide	PataDe Cabra	pide	PataDe Cabra			

	Х	1
cus@Abra	b	1
	а	2

```
class Abra {
    int a = 2, b = 1
0    fun cus(int x): int {
1         a = b + x
2         return pide(a)
      }
    fun pide(int y): int {
3         return a - y * b
      }
}
```

ho

Abra

Cad abra

cus

po		cir			
cus	PataDe Cabra	cus	PataDe Cabra		
pide	PataDe Cabra	pide	PataDe Cabra		

	х	1
cus@Abra	b	1
	а	2⁄2

```
class Abra {
    int a = 2, b = 1
0    fun cus(int x): int {
1         a = b + x
2         return pide(a)
      }
    fun pide(int y): int {
3         return a - y * b
      }
}
```

ho

Abra

Cad abra

cus

ро				cir			
	cus	PataDe Cabra		cus	PataDe Cabra		
	pide	PataDe Cabra		pide	PataDe Cabra		

	Х	1
cus@Abra	b	1
	а	2⁄2

```
class Cadabra extends Abra {
   Abra zo = new PataDeCabra()
0   fun pide(int y): int {
1     return zo.cus(a + b) - y
   }
}
```

ZO	
cus Pata Cabr	
pide Pata Cabr	

		у	2
	pide@Cad abra	zo	Tipo PataDCabra
		b	1
		а	2′2
		х	1
	cus@Abra	b	1
		а	2⁄2

ho		ро			cir		
cus	Abra		cus	PataDe Cabra		cus	PataDe Cabra
pide	Cad abra		pide	PataDe Cabra		pide	PataDe Cabra

```
class Cadabra extends Abra {
   Abra zo = new PataDeCabra()
0   fun pide(int y): int {
1     return zo.cus(a + b) - y
   }
}
```

ZO	
cus Pat	taDe bra
pide Pat	taDe bra

		у	2
	pide@Cad abra	zo	Tipo PataDCabra
		b	1
		а	2′2
	cus@Abra	х	1
		b	1
		а	2⁄2

ho			ро			cir		
cus	Abra		cus	PataDe Cabra		cus	PataDe Cabra	
pide	Cad abra		pide	PataDe Cabra		pide	PataDe Cabra	

									х	3
			a extends	s Cadabı	ra {			cus@Pata	c (de zo)	6
i 0	a = x - 3 $c = a + b * c$							DeCabra	b (de zo)	7
1									a (de zo)	2
3			(a * b + x	<u>:</u>)					у	2
}	}					ZO		nido@Cod	ZO	Tipo PataDCabra
4	_	·				cus	PataDe Cabra	pide@Cad abra		<u> </u>
5	retu	rn c - y '	a		-	Cabia		abia	b	1
}						oide	PataDe Cabra		а	2′ 2
h	ıo	p	O	C	ir				Х	1
cus	Abra	cus	PataDe Cabra	cus	PataDe Cabra	;		cus@Abra	b	1
oide	Cad abra	pide	PataDe Cabra	pide	PataDe Cabra	:			а	2⁄2

									х	3
			ı extends	Cadab	ra {			cus@Pata	c (de zo)	6
		7, c = 6 us(int x):	int {					DeCabra	b (de zo)	7
1 2		x - 3 a + b * c							a (de zo)	20
3			a * b + x)					у	2
} 4 fun pide(int y): int {						ZO cus PataE		pide@Cad	zo	Tipo PataDCabra
5	•	n c - y *					Cabra	abra	b	1
}						pide	PataDe Cabra		а	2′2
h	0	po		c	ir				х	1
cus	Abra	cus	PataDe Cabra	cus	PataDe Cabra	•		cus@Abra	b	1
pide	Cad abra	pide	PataDe Cabra	pide	PataDe Cabra	-			а	2⁄2

									х	3
		DeCabra	extends	Cadabr	a {			cus@Pata	c (de zo)	g [′] 42
		7, c = 6 us(int x):	int {					DeCabra	b (de zo)	7
1 2		x - 3 a + b * c							a (de zo)	20
3		ırn pide(a * b + x)					у	2
} 4	fun pi	ide(int y)	· int {		Γ	ZO cus PataDe		pide@Cad	zo	Tipo PataDCabra
5	-	rn c - y *					Cabra	abra	b	1
}						pide	PataDe Cabra		а	2′2
h	.O	po		ci	ir				х	1
cus	Abra	cus	PataDe Cabra	cus	PataD Cabra			cus@Abra	b	1
pide	Cad abra	pide	PataDe Cabra	pide	PataD Cabra				а	2⁄2

i	nt b = fun c a =	: 7, us(x -	c = 6 (int x):	extends	cadab	ora {			cus@Pata DeCabra	x c (de zo) b (de zo) a (de zo)	3 6 42 7 2 0
3				a * b + x) a*b+	x =0*7				у	2
} 4	fun n	ide	e(int y):	· int {		[cus	D PataDe	pide@Cad	zo	Tipo PataDCabra
5	_		c - y * a				Cabra		abra	b	1
}							pide	PataDe Cabra		а	2′ 2
h	O		ро		(cir				х	1
cus	Abra		cus	PataDe	cus	PataE			cus@Abra	b	1
pide	Cabra			De			а	2⁄2			

ZO

PataDe

PataDe

Cabra

Cabra

```
class PataDeCabra extends Cadabra {
  int b = 7, c = 6
  fun cus(int x): int {
     a = x - 3
c = a + b * c
     return pide(a * b + x)
   fun pide(int y): int {
                                         cus
    return c - y * a
                                         pide
```

ho

Abra

Cad abra

cus

у	3
х	3
c (de zo)	g [′] 42
b (de zo)	7
a (de zo)	20
	c (de zo)

	po		ci	r
	cus	PataDe Cabra	cus	PataDe Cabra
	pide	PataDe Cabra	pide	PataDe Cabra

```
class PataDeCabra extends Cadabra {
   int b = 7, c = 6
    fun cus(int x): int {
       a = x - 3
 c = a + b * c
       return pide(a * b + x)
                                                    ZO
     fun pide(int y): int {
                                                       PataDe
                                              cus
                                                       Cabra
      return c - y * a
                          Retorna
                                              pide
                                                       PataDe
                          42-3*0= 42
                                                       Cabra
                                      cir
                  po
   ho
                       PataDe
                                         PataDe
               cus
                                 cus
cus
      Abra
                       Cabra
                                         Cabra
pide
      Cad
               pide
                       PataDe
                                 pide
                                         PataDe
      abra
```

Cabra

Cabra

pide@Pat aDeCabra	у	3
	х	3
cus@Pata	c (de zo)	g [′] 42
DeCabra	b (de zo)	7
	a (de zo)	20

i	nt b = fun c a =	7, us(x -	c = 6 (int x):	extends int {	Cadabı	a {			cus@Pata DeCabra	x c (de zo) b (de zo) a (de zo)	3 6 42 7 2 0
3				a * b + x)	Retorn	a 42	Z	n		у	2
} 4	fun n	ide	e(int y):	: int {			cus	PataDe	pide@Cad	ZO	Tipo PataDCabra
5			c - y * a					Cabra	abra	b	1
}							pide PataDe Cabra			а	2′ 2
h	0		po		C	ir				х	1
cus	Abra		cus	PataDe Cabra	cus	Pata E Cabra			cus@Abra	b	1
pide	Cad abra		pide	PataDe Cabra	pide	Pata[Cabra	De De			а	2⁄2

```
class Cadabra extends Abra {
    Abra zo = new PataDeCabra()
     fun pide(int y): int {
       return zo.cus(a + b) - y Retorna 42 -2=40 zo
                                                                                        Tipo PataDCabra
                                                                           ZO
                                                              pide@Cad
                                                    PataDe
                                            cus
                                                    Cabra
                                                              abra
                                                                           b
                                            pide
                                                    PataDe
                                                                                       2′2
                                                    Cabra
                                                                           a
                                                                           X
                                     cir
                  po
   ho
                                                                           b
                                                              cus@Abra
                      PataDe
                                        PataDe
              cus
                                cus
cus
     Abra
                      Cabra
                                        Cabra
                                                                                       2/2
```

pide

Cad

abra

pide

PataDe

Cabra

pide

PataDe

Cabra

a

```
class Abra {
   int a = 2, b = 1
0   fun cus(int x): int {
1      a = b + x
2      return pide(a)      Retorna 40
   }
   fun pide(int y): int {
3      return a - y * b
   }
}
```

ho

Abra

Cad abra

cus

po		cir					
cus	PataDe Cabra		cus	PataDe Cabra			
pide	PataDe Cabra		pide	PataDe Cabra			

cus@Abra	х	1
	b	1
	а	2⁄2

$$print(40 + po.cus(1) + cir.cus(1))$$

h	0	po		ci	r
cus	Abra	cus	PataDe Cabra	cus	PataDe Cabra
pide	Cad abra	pide	PataDe Cabra	pide	PataDe Cabra

```
class PataDeCabra extends Cadabra {
   int b = 7, c = 6
0   fun cus(int x): int {
1      a = x - 3
2      c = a + b * c
3      return pide(a * b + x)
   }
4   fun pide(int y): int {
5      return c - y * a
}
```

ho

Abra

Cad abra

cus

ро		ci	r	
cus	PataDe Cabra	cus	PataDe Cabra	
pide	PataDe Cabra	pide	PataDe Cabra	

ZO					
cus	PataDe Cabra				
pide	PataDe Cabra				

	cus@PataD eCabra	х	1
		c (de po)	6
		b (de po)	7
		a (de po)	2

```
class PataDeCabra extends Cadabra {
   int b = 7, c = 6
0   fun cus(int x): int {
1         a = x - 3
2         c = a + b * c
3         return pide(a * b + x)
      }
4   fun pide(int y): int {
5       return c - y * a
}
```

ho

Abra

Cad abra

cus

po		ci	r	
cus	PataDe Cabra	cus	PataDe Cabra	
pide	PataDe Cabra	pide	PataDe Cabra	

ZO	1		X	1
cus	PataDe	eCabra	c (de po)	6
	Cabra		b (de po)	7
pide	e PataDe Cabra		a (de po)	2′-2

```
class PataDeCabra extends Cadabra {
   int b = 7, c = 6
0   fun cus(int x): int {
1      a = x - 3
2      c = a + b * c     c = -2 + 7*6=40
3      return pide(a * b + x)
   }
4   fun pide(int y): int {
5      return c - y * a
}
```

ho

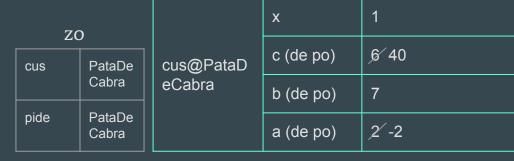
Abra

Cad

abra

cus

po		ci	r	
cus	PataDe Cabra	cus	PataDe Cabra	
pide	PataDe Cabra	pide	PataDe Cabra	



5 }	return c - y ^ a										
							ZO			х	1
ho		po	po		cir		cus PataDe		cus@PataD	c (de po)	6⁄40
cus	Abra	cus	PataDe Cabra	cus	PataDe Cabra			Cabra	eCabra	b (de po)	7
pide	Cad abra	pide	PataDe Cabra	pide	PataDe Cabra	pic	de	PataDe Cabra		a (de po)	2′-2

class PataDeCabra extends Cadabra {

```
int b = 7, c = 6
    fun cus(int x): int {
       a = x - 3
     c = a + b * c
       return pide(a * b + x)
     fun pide(int y): int {
                                                                                                      -13
      return c - y * a
                                                                          pide@PataD
                                                                          eCabra
                                                             ZO
                                        cir
                   po
                                                                                                      6/40
   ho
                                                                                         c (de po)
                                                                          cus@PataD
                                                                PataDe
                                                        cus
                        PataDe
                                           PataDe
                                                                Cabra
               cus
                                   cus
                                                                          eCabra
cus
      Abra
                                                                                         b (de po)
                        Cabra
                                           Cabra
                                                        pide
                                                                PataDe
pide
      Cad
                                                                                                      2'-2
                                                                                        a (de po)
               pide
                        PataDe
                                   pide
                                           PataDe
                                                                Cabra
      abra
                        Cabra
                                           Cabra
```

class PataDeCabra extends Cadabra {

```
int b = 7, c = 6
    fun cus(int x): int {
       a = x - 3
 2 c = a + b * c
       return pide(a * b + x)
     fun pide(int y): int {
                                                                                                     -13
      return c - y * a  c-y^*a = 40-(-13)^*(-2)=14 
                                                                         pide@PataD
                                                                         eCabra
                                                             ZO
                                       cir
                   po
                                                                                        c (de po)
                                                                                                     6/40
   ho
                                                                         cus@PataD
                                                                PataDe
                                                       cus
                        PataDe
                                           PataDe
                                                                Cabra
               cus
                                  cus
                                                                         eCabra
cus
      Abra
                                                                                        b (de po)
                        Cabra
                                           Cabra
                                                       pide
                                                                PataDe
pide
      Cad
                                                                                                     2'-2
                                                                                        a (de po)
               pide
                        PataDe
                                  pide
                                           PataDe
                                                                Cabra
      abra
                        Cabra
                                           Cabra
```

```
class PataDeCabra extends Cadabra {
   int b = 7, c = 6
0   fun cus(int x): int {
1         a = x - 3
2         c = a + b * c
3         return pide(a * b + x)         pide(a*b+x) = 14
      }
4   fun pide(int y): int {
5       return c - y * a
}
```

-								
ho			po		cir			
cus	Abra		cus	PataDe Cabra		cus	PataDe Cabra	
oide	Cad abra		pide	PataDe Cabra		pide	PataDe Cabra	

ZO)		х	1
cus	PataDe	cus@PataD eCabra	c (de po)	6´40
	Cabra		b (de po)	7
pide	PataDe Cabra		a (de po)	2′-2

print(
$$\frac{40}{40} + \frac{14}{14} + \text{cir.cus}(1)$$
)

h	o	ро			cir		
cus	Abra	cus	PataDe Cabra		cus	PataDe Cabra	
pide	Cad abra	pide	PataDe Cabra		pide	PataDe Cabra	

```
class PataDeCabra extends Cadabra {
  int b = 7, c = 6
0  fun cus(int x): int {
1    a = x - 3
2    c = a + b * c
3   return pide(a * b + x)
  }
4  fun pide(int y): int {
5   return c - y * a
}
```

ho

Abra

Cad abra

cus

po		ci	r	
cus	PataDe Cabra	cus	PataDe Cabra	
pide	PataDe Cabra	pide	PataDe Cabra	

	ZO)		х	1
	cus	PataDe	eCabra	c (de cir)	6
		Cabra		b (de cir)	7
	pide	PataDe Cabra	a (de cir)	2	
١					

```
class PataDeCabra extends Cadabra {
  int b = 7, c = 6
0  fun cus(int x): int {
1    a = x - 3
2    c = a + b * c
3   return pide(a * b + x)
  }
4  fun pide(int y): int {
5   return c - y * a
```

ho

Abra

Cad abra

cus

ро		ci	r	
cus	PataDe Cabra	cus	PataDe Cabra	
pide	PataDe Cabra	pide	PataDe Cabra	

ZC	,		х	1
cus	PataDe Cabra	cus@PataD eCabra	c (de cir)	6
			b (de cir)	7
pide	PataDe Cabra		a (de cir)	2′-2

```
class PataDeCabra extends Cadabra {
   int b = 7, c = 6
0   fun cus(int x): int {
1      a = x - 3
2      c = a + b * c     c = -2 + 7*6=40
3      return pide(a * b + x)
   }
4   fun pide(int y): int {
5      return c - y * a
}
```

ho

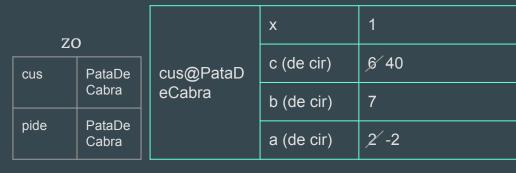
Abra

Cad

abra

cus

po		cir				
cus	PataDe Cabra	cus	PataDe Cabra			
pide	PataDe Cabra	pide	PataDe Cabra			



ho			ро			cir		
cus	Abra		cus	PataDe Cabra		cus	PataDe Cabra	
oide	Cad abra		pide	PataDe Cabra		pide	PataDe Cabra	

ZC)	
cus	PataDe Cabra	cus@ eCabi
pide	PataDe Cabra	

	х	1
cus@PataD	c (de cir)	6∕40
eCabra	b (de cir)	7
	a (de cir)	2′-2

```
class PataDeCabra extends Cadabra {
   int b = 7, c = 6
0   fun cus(int x): int {
1      a = x - 3
2      c = a + b * c
3      return pide(a * b + x)
   }
4   fun pide(int y): int {
5      return c - y * a
}
```

cus

)	po			cir
Abra	cus	PataDe Cabra	cus	PataDe Cabra
Cad abra	pide	PataDe Cabra	pide	PataDe Cabra

class PataDeCabra extends Cadabra {

pide

abra

PataDe

Cabra

pide

PataDe

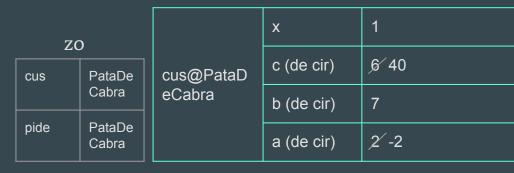
Cabra

```
int b = 7, c = 6
    fun cus(int x): int {
       a = x - 3
 2 c = a + b * c
       return pide(a * b + x)
     fun pide(int y): int {
                                                                                                   -13
      return c - y * a -y^*a = 40-(-13)*(-2)=14
                                                                        pide@PataD
                                                                        eCabra
                                                           ZO
                                       cir
                   po
                                                                                      c (de cir)
                                                                                                   6/40
   ho
                                                                        cus@PataD
                                                              PataDe
                                                      cus
                       PataDe
                                          PataDe
                                                              Cabra
               cus
                                  cus
                                                                        eCabra
cus
      Abra
                                                                                      b (de cir)
                       Cabra
                                          Cabra
                                                      pide
                                                               PataDe
pide
      Cad
                                                                                                   2'-2
                                                                                      a (de cir)
```

Cabra

			no		ci	r	
10		ро		cir			
	Abra		cus	PataDe Cabra	cus	PataDe Cabra	
	Cad abra		pide	PataDe Cabra	pide	PataDe Cabra	

cus



```
Abra ho = new Cadabra()
```

Abra po = new PataDeCabra()

Cadabra cir = new PataDeCabra()

print(ho.cus(1) + po.cus(1) + cir.cus(1))

ho			ро			cir		
cus	Abra		cus	PataDe Cabra		cus	PataDe Cabra	
pide	Cad abra		pide	PataDe Cabra		pide	PataDe Cabra	

Imprime 68

¡Un héroe es alguien que supera cada

obstáculo que la vida pone en su camino!