

Esta prova tem a duração de **2 horas** e é **sem consulta**.

1. Considerando os programas da figura 1, comente as seguintes declarações, *justificando-as*.

- (a) A classe *T* pertence à package *pack*;
- (b) Todas as classes da package *pack*, são acessíveis em *T*;
- (c) Todas as instâncias da classe *X* têm o mesmo valor para *a*;
- (d) A acessibilidade da variável *v*, é restricta aos ficheiros da package *pack*;
- (e) A classe *X* tem dois métodos constructores;
- (f) O métodos das linhas 25. e 29., têm a mesma assinatura;
- (g) O valor de retorno do método da linha 7. é *void*;
- (h) O return da linha 42. é desnecessário;
- (i) O resultado da linha 7. na classe *T* é *true*;
- (j) A variável *a*, deveria ser precedida dos modificadores *static final*;
- (k) A mensagem da linha 5. na classe *T* é incorreta;
- (l) A mensagem da linha 8. na classe *T* é incorreta;
- (m) A mensagem da linha 10. na classe *T* é correta;
- (n) O resultado da linha 14. é (0;30;0;0;0;0;0;0;0;0).

2. Implemente a package *Logical*, constituída por:

- (a) A classe *Bit*, cuja função é armazenar a informação contida num bit, i.e. 0 ou 1, ou, uma representação equivalente. Implemente métodos:
 - Constructores;
 - selectores;
 - modificadores;
 - para realizar as operações And, Or e Not. O retorno destes métodos deverá ser um bit.

- (b) A classe *Byte*, correspondente ao agrupamento de 8 bits. Implemente:

7	6	5	4	3	2	1	0
0	1	0	1	1	0	1	0

BIT DE ORDEM 5

- Um constructor em que o objecto criado é um byte com todos os bits a zero;
- Um construtor, em que sendo dado um array de 0's e 1's, é criado o byte correspondente;
- O método *toString*;
- O método *bit(n)*, que devolve o bit de ordem *n*;
- O método *setBit(n,b)*, que afecta o bit de ordem *n*, com o bit *b*;
- O método *setBit(n,j)*, que afecta o bit de ordem *n*, com o valor *j*;
- O método *shiftLeft(n)*, que permite introduzir *n*-bits 0, nas *n* posições menos significativas do byte.

<pre> 1. package pack; 2. public class X{ 3. static boolean p; 4. int a=10; 5. int[] v; 6. 7. public X(int n){ 8. a=n; 9. ini(); 10. setP(); 11. 12. public X(){ 13. ini(-8); 14. setP(); 15. } 16. 17. public static void setP(){ 18. p=true; 19. } 20. 21. public static void resetP(){ 22. p=false; 23. } 24. 25. private void ini(){ 26. v=new int[a]; 27. } 28. 29. private void ini(int x){ 30. if (x>0) 31. v=new int[x]; 32. else 33. v=new int[a]; 34. } 35. 36. public int put(int j, int i){ 37. if (p) 38. v[j]= i; 39. else 40. v[j] = v[j]+ i; 41. 42. return v[j]; 43. } 44. 45. public String toString(){ 46. String r="(" ; 47. for(int i=0; i<v.length;i++) 48. r=r+v[i]+ " "; 49. return r +")"; 50. } 51. } </pre>	<pre> 1. import pack.X; 2. public class T{ 3. public static void main(String[] args){ 4. X ia=new X(); 5. ia.ini(); 6. X i1=new X(10); 7. System.out.println(ia1==ia); 8. ia.setP(); 9. ia1.put(0,10); 10. X.resetP(); 11. ia.put(1,20); 12. ia.put(1,10); 13. System.out.println(ia1); 14. System.out.println(ia.toString()); 15. } 16. } </pre>
--	--

Figure 1: X.java à esquerda e T.java à direita.