Ejemplos de Algoritmos

Vilchis Salazar Jose Antonio

Algoritmo 1

Dado un arreglo de enteros, invertir sus valores internos. Imprime el arreglo antes de ejecutar el algoritmo, y después de la inversión.

Código

```
1  public static void exchange(int[] array) {
2    int aux, index, mid;
3    mid = array.length / 2;
4    print(array);
5    for(int i = 0; i < mid; i++) {
6      index = array.length - i - 1;
7      aux = array[i];
8      array[i] = array[index];
9      array[index] = aux;
10   }
11   print(array);
12 }
```

Estado de variables

```
1  ... exchange(int[] array) {
2    int aux, index, mid;
3    mid = array.length / 2;
4    print(array);
5    for(int i = 0; i < mid; i++) {
6  ...
7  }
```

array	i	index	mid	aux
[1 2 3 4]	0	-	2	-

Estado de variables

```
1  for(int i = 0; i < mid; i++) {
2    index = array.length - i - 1;
3    aux = array[i];
4    array[i] = array[index];
5    array[index] = aux;
6  }
```

array	i	index	mid	aux
[1 2 3 4]	0	-	2	-
[4 2 3 1]	1	3	2	1

Estado de variables

```
1  for(int i = 0; i < mid; i++) {
2    index = array.length - i - 1;
3    aux = array[i];
4    array[i] = array[index];
5    array[index] = aux;
6  }
```

array	i	index	mid	aux
[1 2 3 4]	0	-	2	-
[4 2 3 1]	1	3	2	1
[4 3 2 1]	2	2	2	2

Algoritmo 2

Imprime la generación de secuencias de longitud n con d dígitos.

Código

```
1  public static void sequences(int d, int l) {
2    int array[] = new int[l];
3    boolean next;
4    do {
5      print(array);
6      next = next(d, l - 1, array);
7    } while(next);
8  }
```

Código

```
1  public static boolean next(int d, int i, int array[]) {
2    if(i == -1) {
3      return false;
4    }
5    array[i]++;
6    if(array[i] == d) {
7      array[i] = 0;
8      return next(d, i - 1, array);
9    } else {
10     return true;
11   }
12 }
```

Estado de variables

```
1  do {
2    print(array);
3    next = next(d, l - 1, array);
4  } while(next);
```

array	next	d	l - 1
[0 0 0 0]	-	5	3

Estado de variables

```
1  ... next(int d, int i, int array[]) {
2    if(i == -1) {
3      return false;
4    }
5    array[i]++;
6    if(array[i] == d) {
7      array[i] = 0;
8      return next(d, i - 1, array);
9    } else {
10     return true;
11   }
12 }
```

array	i	d	return
[0 0 0 0]	3	5	-
[0 0 0 1]	3	5	-
[0 0 0 1]	3	5	T

Estado de variables

```
1  do {
2    print(array);
3    next = next(d, l - 1, array);
4  } while(next);
```

array	next	d	l - 1
[0 0 0 0]	-	5	3
[0 0 0 1]	T	5	3

Estado de variables

```
1  ... next(int d, int i, int array[]) {
2    if(i == -1) {
3      return false;
4    }
5    array[i]++;
6    if(array[i] == d) {
7      array[i] = 0;
8      return next(d, i - 1, array);
9    } else {
10     return true;
11   }
12 }
```

array	i	d	return
[0 0 0 1]	3	5	-
[0 0 0 2]	3	5	-
[0 0 0 2]	3	5	T

Estado de variables

```
1  do {
2    print(array);
3    next = next(d, l - 1, array);
4  } while(next);
```

array	next	d	l - 1
[0 0 0 0]	-	5	3
[0 0 0 1]	T	5	3
[0 0 0 2]	T	5	3

Estado de variables

```
1  ... next(int d, int i, int array[]) {
2    if(i == -1) {
3      return false;
4    }
5    array[i]++;
6    if(array[i] == d) {
7      array[i] = 0;
8      return next(d, i - 1, array);
9    } else {
10     return true;
11   }
12 }
```

array	i	d	return
[0 0 0 2]	3	5	-
[0 0 0 3]	3	5	-
[0 0 0 3]	3	5	T

Estado de variables

```
1  do {
2    print(array);
3    next = next(d, l - 1, array);
4  } while(next);
```

array	next	d	l - 1
[0 0 0 0]	-	5	3
[0 0 0 1]	T	5	3
[0 0 0 2]	T	5	3
[0 0 0 3]	T	5	3

Estado de variables

```
1  ... next(int d, int i, int array[]) {
2    if(i == -1) {
3      return false;
4    }
5    array[i]++;
6    if(array[i] == d) {
7      array[i] = 0;
8      return next(d, i - 1, array);
9    } else {
10     return true;
11   }
12 }
```

array	i	d	return
[0 0 0 3]	3	5	-
[0 0 0 4]	3	5	-
[0 0 0 4]	3	5	T

Estado de variables

```
1  do {
2    print(array);
3    next = next(d, l - 1, array);
4  } while(next);
```

array	next	d	l - 1
[0 0 0 0]	-	5	3
[0 0 0 1]	T	5	3
[0 0 0 2]	T	5	3
[0 0 0 3]	T	5	3
[0 0 0 4]	T	5	3

Estado de variables

```
1  ... next(int d, int i, int array[]) {
2    if(i == -1) {
3      return false;
4    }
5    array[i]++;
6    if(array[i] == d) {
7      array[i] = 0;
8      return next(d, i - 1, array);
9    } else {
10     return true;
11   }
12 }
```

array	i	d	return
[0 0 0 4]	3	5	-
[0 0 0 5]	3	5	-
[0 0 0 0]	3	5	-

Estado de variables

```
1  ... next(int d, int i, int array[]) {
2    if(i == -1) {
3      return false;
4    }
5    array[i]++;
6    if(array[i] == d) {
7      array[i] = 0;
8      return next(d, i - 1, array);
9    } else {
10     return true;
11   }
12 }
```

array	i	d	return
[0 0 0 0]	2	5	-
[0 0 1 0]	2	5	-
[0 0 1 0]	2	5	T

Estado de variables

```
1  do {
2    print(array);
3    next = next(d, l - 1, array);
4  } while(next);
```

array	next	d	l - 1
[0 0 0 0]	-	5	3
[0 0 0 1]	T	5	3
[0 0 0 2]	T	5	3
[0 0 0 3]	T	5	3
[0 0 0 4]	T	5	3
[0 0 1 0]	T	5	3

Estado de variables

Ultima llamada.

```
1  do {
2    print(array);
3    next = next(d, l - 1, array);
4  } while(next);
```

array	next	d	l - 1
[4 4 4 4]	-	5	3

Estado de variables

```
1  ... next(int d, int i, int array[]) {
2    if(i == -1) {
3      return false;
4    }
5    array[i]++;
6    if(array[i] == d) {
7      array[i] = 0;
8      return next(d, i - 1, array);
9    } else {
10     return true;
11   }
12 }
```

array	i	d	return
[4 4 4 4]	3	5	-
[4 4 4 5]	3	5	-
[4 4 4 0]	3	5	-

Estado de variables

```
1  ... next(int d, int i, int array[]) {
2    if(i == -1) {
3      return false;
4    }
5    array[i]++;
6    if(array[i] == d) {
7      array[i] = 0;
8      return next(d, i - 1, array);
9    } else {
10     return true;
11   }
12 }
```

array	i	d	return
[4 4 4 0]	2	5	-
[4 4 5 0]	2	5	-
[4 4 0 0]	2	5	-

Estado de variables

```
1  ... next(int d, int i, int array[]) {
2    if(i == -1) {
3      return false;
4    }
5    array[i]++;
6    if(array[i] == d) {
7      array[i] = 0;
8      return next(d, i - 1, array);
9    } else {
10     return true;
11   }
12 }
```

array	i	d	return
[4 4 0 0]	1	5	-
[4 5 0 0]	1	5	-
[4 0 0 0]	1	5	-

Estado de variables

```
1  ... next(int d, int i, int array[]) {
2    if(i == -1) {
3      return false;
4    }
5    array[i]++;
6    if(array[i] == d) {
7      array[i] = 0;
8      return next(d, i - 1, array);
9    } else {
10     return true;
11   }
12 }
```

array	i	d	return
[4 0 0 0]	0	5	-
[5 0 0 0]	0	5	-
[0 0 0 0]	0	5	-

Estado de variables

```
1  ... next(int d, int i, int array[]) {
2    if(i == -1) {
3      return false;
4    }
5    array[i]++;
6    if(array[i] == d) {
7      array[i] = 0;
8      return next(d, i - 1, array);
9    } else {
10     return true;
11   }
12 }
```

array	i	d	return
[0 0 0 0]	-1	5	F

Estado de variables

```
1  do {
2    print(array);
3    next = next(d, l - 1, array);
4  } while(next);
```

array	next	d	l - 1
[0 0 0 0]	F	5	3