

Esercizi sulle tecniche di ordinamento

(Fondamenti di Informatica 2 – Walter Didimo)

Soluzioni

Esercizio 2 Il Bucket Sort è una tecnica di ordinamento di sequenze di numeri naturali che può risultare più efficiente di ogni altra, a patto di assumere che tutti i numeri della sequenza siano non più grandi di una certa costante M . Assumendo di chiamare con v l'array di naturali da ordinare e con s l'array ordinato che contiene gli stessi elementi di v , la tecnica opera nel modo seguente:

- 1) Predisporre un array *bucket* di interi di dimensione $M+1$; il valore di *bucket*[j] dirà quante volte il numero j è presente nell'array v da ordinare.
- 2) Inizializza tutti gli elementi dell'array *bucket* con il valore 0.
- 3) Scandisce l'array v una sola volta, dal primo all'ultimo elemento; ogni volta che viene visitato un elemento $v[i]$, viene incrementato di una unità il valore di *bucket*[$v[i]$].
- 4) Inizializza la posizione corrente di s con il valore 0. Scandisce l'array *bucket* dal primo all'ultimo elemento, e ogni volta che visita un elemento *bucket*[j] con valore superiore a 0, aggiunge ad s , a partire dalla sua posizione corrente, il valore j ripetuto un numero di volte pari a *bucket*[j].

Si chiede di:

- Implementare il seguente metodo di classe:

```
/* restituisce un array ottenuto ordinando l'array v
PRE: gli elementi di v assumono valori nell'intervallo [0,M]*/
public static int[] bucketSort (int[] v, int M)
```

- Verificare il corretto funzionamento del metodo implementato

Soluzione

```
public class Ordinamento {

    public static int[] bucketSort (int[] v, int M){
        int[] s = new int[v.length];
        int[] bucket;

        // istanzia i buckets e li inizializza a false
        bucket = new int[M+1];

        // riempie i bucket
        for (int i=0; i<v.length; i++){
            int j = v[i];
            bucket[j]++;
        }
        // costruisce l'array ordinato
        int i=0;
        for (int j=0; j<bucket.length; j++)
            for (int h=0; h<bucket[j]; h++)
                {
                    s[i]=j;
                    i++;
                }
        return s;
    }
}
```

```
public static void main (String[] args){
    InputWindow in = new InputWindow ();
    int dim = in.readInt ("Numero di elementi da ordinare?");
    int M = 0;
    int[] v = new int[dim];
    for (int i=0; i<v.length; i++){
        v[i] = in.readInt ("Elemento " + i + "?");
        if (M < v[i])
            M = v[i];
    }
    int[] s = Ordinamento.bucketSort (v,M);
    System.out.println ("Sequenza ordinata:");
    for (int i=0; i<s.length; i++)
        System.out.print (s[i] + " ");
    System.out.println ();
}
}
```