



ESCUELA UNIVERSITARIA POLITÉCNICA  
Departamento de Ciencias Politécnicas  
Grado en Ingeniería Informática

Prácticas Algoritmia  
Boletín 4. Algoritmos de búsqueda

Curso 2017/2018

Profesor: Andrés Muñoz

# Boletín 4. Algoritmos de búsqueda

---

## 1. Objetivo

El objetivo de este pretende que los alumnos analicen, comprendan, apliquen y sean capaces de explicar los algoritmos de búsquedas de patrones KMP y BM.

## 2. Ejercicios

1. **(2,5 puntos)** Realizar un algoritmo para realizar una búsqueda de patrones mediante búsqueda directa. El algoritmo debe devolver todas las posiciones del texto original donde comienza el patrón buscado.
2. **(2,5 puntos)** Explicar **detalladamente** qué hace el código de las funciones **preKMP()** y **KMPSearch()** de la implementación del algoritmo KMP proporcionado por el profesor en el fichero *kmp.cpp*. Se debe utilizar al menos un ejemplo sencillo para demostrar su funcionamiento.
3. **(2,5 puntos)** Completar el código de la función **main()** de los ficheros *km.cpp* y *bm.cpp* proporcionados por el profesor para buscar el patrón **“porompomrom”** en el fichero *“textobusq.txt”* que se encuentra en la misma carpeta del CV de este boletín. Antes de ejecutar los programas, calcular las tablas “siguiente” para KMP y D1, D2 para BM. Una vez calculadas, ejecutar los programas. ¿Para el caso de KMP, obtienes los mismos resultados? En caso contrario, indicar por qué. Finalmente, mostrar todas las posiciones del fichero donde se encuentra la primera letra del patrón buscado.
4. **(2.5 puntos)** Comparar el tiempo de ejecución de los tres algoritmos (búsqueda directa, KMP y BM) y el número de veces que se reinicia la búsqueda de un patrón **y comentar los resultados obtenidos, en relación con los resultados teóricos vistos en clase**.  
Nota: en el caso de KMP y BM, calcular el tiempo de ejecución teniendo en cuenta la creación de las tablas “siguiente” en KMP y D1, D2 en BM, más el tiempo de ejecución del algoritmo de búsqueda concreto.

### **Ejercicio opcional (Se sumará 0,5 punto si es correcto a la nota final de prácticas):**

Explicar **detalladamente** qué hace el código de las funciones **prepare\_badcharacter\_heuristic()**, **prepare\_goodsuffix\_heuristic()** y **boyermoore\_search()** de la implementación del algoritmo BM proporcionado por el profesor en el fichero *bm.cpp*. y debe utilizar al menos un ejemplo sencillo para demostrar su funcionamiento. ¿Se calculan las tablas D1 y D2 de la misma forma que se explican en las diapositivas del tema 4? En caso de que no sea así, ¿qué diferencias observas?

### 3. Entregables y puntuación

- Memoria que contenga las respuestas e información pedida en los ejercicios del boletín (NO INCLUIR EL CÓDIGO DE LOS PROGRAMAS). **Es especialmente importante responder correctamente a las preguntas de los ejercicios para una buena valoración del boletín.**
- Un proyecto de Dev-C++ que incluya los ficheros .c con el código de los programas y que esté listo para ser compilado y ejecutado (NO ENTREGAD LOS FICHEROS DE CADA EJERCICIO POR SEPARADO O NO SE CORREGIRÁN LAS PRÁCTICAS).
- **Fecha de entrega: 11 de Febrero 2018**