



Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

Laboratorios de computación salas A y B

Tista Garcia Edgar

Profesor:

Estructura de Datos y Algoritmos II

Asignatura:

5

Grupo:

2

No de Práctica(s):

Calzada Martinez Jonathan Omar

Integrante(s):

*No. de Equipo de
cómputo empleado*

35

2019-2

Semestre:

Fecha de entrega:

Obervaciones:

CALIFICACIÓN: _____

a) Con respecto al enfoque visto en clase en realidad no cambia ya que el algoritmo es el mismo. Se utiliza un for hasta el tamaño del arreglo y disminuye 1 para ir comparando uno por uno. Y se coloca un while para hacer el intercambio.

En la imagen de abajo se puede ver en que momento se hace la iteración.

```
while ( ( i > -1) && ( array [i] > key ) ) {  
    array [i+1] = array [i];  
    i--;  
}  
array[i+1] = key; | ←
```

```
run:  
Arreglo Original  
10 15 4 3 44 12 59 23  
Arreglo ordenado  
3 4 10 12 15 23 44 59  
BUILD SUCCESSFUL (total time: 0 seconds)  
|
```

b)

El en InsertionSort solo se le mandanda el arreglo para que pueda hacer el ordenamiento, a diferencia de selectionSort que de igual manera se le manda el arreglo a la funcion pero tambien se llama a un modulo para que ejecute la funcion.

```
//InsertionSort.insertionSort(arr2);  
SelectionSort seleccion = new SelectionSort();  
seleccion.selectionSort(arr1);
```

```

run:
Arreglo Original
10 15 4 3 44 12 59 23
Iteración 1
Índice del mínimo 3
Valor 2
2 14 3 9 43 11 58 22
Iteración 2
Índice del mínimo 2
Valor 3
2 3 14 9 43 11 58 22
Iteración 3
Índice del mínimo 3
Valor 9
2 3 9 14 43 11 58 22
Iteración 4
Índice del mínimo 5
Valor 11
2 3 9 11 43 14 58 22
Iteración 5
Índice del mínimo 5
Valor 14
2 3 9 11 14 43 58 22
Iteración 6
Índice del mínimo 7
Valor 22
2 3 9 11 14 22 58 43
Iteración 7
Índice del mínimo 7
Valor 43
2 3 9 11 14 22 43 58
Arreglo ordenado
10 15 4 3 44 12 59 23
BUILD SUCCESSFUL (total time: 3 seconds)

```

c) Con respecto al enfoque visto en clase no hay diferencia, trabaja el algoritmo de la misma manera. Se utiliza un padre y sobre ese va trabajando y haciendo los cambios. Con respecto al diagrama del árbol binario, trabaja con los padres en donde comienza con el 0 luego con el 1 y así sucesivamente compara el mayor de los hijos y de esta manera los acomoda en el nuevo arreglo y vuelve a hacer la iteración.

d) A diferencia de C, en java no empizan con un encabezado en donde se encuentran las bibliotecas principales. Se muestra una clase y en ella trabaja como si fuese la funcion principal, comparandolo con C con el (Int main) se encuentra una funcion en donde se utiliza para imprimir el arreglo que es llamada en cada una de las funciones utilizadas de ordenamiento. Esta funcion esta hecha con public static void main.

Conclusiones

Los algoritmos SelectionSort y InsertionSort a mi parecer son muy parecidos cambian en pocas cosas, en la manera de funcionar.

