UNIVERSIDAD ORT
Uruguay

Facultad de Ingeniería
Bernard Wand-Polak
Cuareim 1451
11.100 Montevideo, Uruguay
Tel 902 15 05 Fax 908 13 70
www.ort.edu.uy

EVALUACION	Solución Examen	**FECHA**	09/02/2022
MATERIA	Algoritmos 1		
CARRERA	Analista Programador / Analista en Tecnologías de la Información		
CONDICIONES	**- Puntos: Máximo: Mínimo:** **- Duración: 2 horas** **- Sin material**		
Nombre	**Nro estudiante**	**Nota**	

Ejercicio 1

Implementar un algoritmo que reciba un vector ordenado y haga una búsqueda binaria
El algoritmo debe ser resuelto en forma **recursiva.**

Utilizar la siguiente firma:

public int buscar(int[] números, int número, int inicio, int fin) {

solución

```
public int buscar(int[] números, int número, int inicio, int fin) {

    int centro = (inicio + fin) / 2;

    if (fin < inicio) {
        return -1;
    }

    if (número < números[centro]) {
        return buscar(números, número, inicio, centro - 1);
    }

    if (número > números[centro]) {
        return buscar(números, número, centro + 1, fin);
    }

    if (número == números[centro]) {
        return centro;
    }

    return -1;
}
```

Facultad de Ingeniería
Bernard Wand-Polak
Cuareim 1451
11.100 Montevideo, Uruguay
Tel 902 15 05 Fax 908 13 70
www.ort.edu.uy

Ejercicio 2

Dado el siguiente vector

int v[] = {64, 34, 25, 12, 22, 11, 90};

a) Indique a que método de ordenamiento corresponde la siguiente secuencia

- 34- 25- 12- 22- 11- 64- 90
- 25- 12- 22- 11- 34- 64- 90
- 12- 22- 11- 25- 34- 64- 90
- 12- 11- 22- 25- 34- 64- 90
- 11- 12- 22- 25- 34- 64- 90
- 11- 12- 22- 25- 34- 64- 90

Solución: Burbuja

b) Implemente el método de ordenamiento en forma recursiva.

```
// Solución
  static void burbuja (int v[], int n)
   {
      // Base case
      if (n == 1)
         return;

      for (int i=0; i<n-1; i++)
         if (v[i] > v[i+1])
         {
            int temp = v[i];
            v[i] = v[i+1];
            v[i+1] = temp;
         }

      Burbuja(v, n-1);
   }
   }
}
```

Facultad de Ingeniería
Bernard Wand-Polak
Cuareim 1451
11.100 Montevideo, Uruguay
Tel 902 15 05 Fax 908 13 70
www.ort.edu.uy

Ejercicio 3

Implemente un método de búsqueda por bipartición en forma recursiva que retorne -1 si no encuentra el valor x recibido como parámetro y si lo encuentra retorne la posición dentro del vector.

Firma a utilizar int busqueda(int v[], int X, int i, int j)

Ejemplo

Int v[] = { 11, 12,22, 25, 34, 64, 90};

System.out.println(busqueda(v, 64, 0, v.length-1)); retorna 5
System.out.println(busqueda(v, 34, 0, v.length-1)); retorna 4
System.out.println(busqueda(v, 200, 0, v.length-1)); retorna -1

```
// Solución
   static int busqueda(int v[], int X, int i, int j) {
      int medio;
      if (i > j) {
         return -1;
      }
      medio = (i + j) / 2;
      if (v[medio] < X) {
         return busqueda(v, X, medio + 1, j);
      } else {
         if (v[medio] > X) {
            return busqueda(v, X, i, medio - 1);
         } else {
            return medio;
         }
      }
   }
```

Facultad de Ingeniería
Bernard Wand-Polak
Cuareim 1451
11.100 Montevideo, Uruguay
Tel 902 15 05 Fax 908 13 70
www.ort.edu.uy

Ejercicio 4

Escribir una función: void Reemplazar(pila P,int nuevo,int viejo)
que tenga como argumentos una pila P de enteros un valor int: nuevo y un valor int viejo de forma que, si el segundo valor aparece en algún lugar de la pila, sea reemplazado.

Solucion

```
void Reemplazar(pila P, int nuevo, int viejo)
{
  pila Q;
  int aux;

  /*Creamos una pila auxiliar para almacenar enteros*/
  Q=CrearPila(sizeof(int));

  /*Vamos sacando elementos de P y los vamos metiendo en Q*/
  /*excepto en el caso de sacar viejo,que insertamos nuevo*/
  while (!VaciaPila(P)){
    Tope(&aux,P);
    if (aux==viejo)
       Push(&nuevo,Q);
    else Push(&aux,Q);
    Pop(P);
  }

  /*ya tenemos en Q el resultado pero al contrario por tanto*/
  /*solo resta volcarla de nuevo en P*/
  while(!VaciaPila(Q)){
    Tope(&aux,Q);
    Pop(Q);
    Push(&aux,P);
  }

  /*Finalmente liberamos los recursos que fueron reservados*/
  /*para la pila auxiliar Q*/

  DestruirPila(Q);

}
```

Facultad de Ingeniería
Bernard Wand-Polak
Cuareim 1451
11.100 Montevideo, Uruguay
Tel 902 15 05 Fax 908 13 70
www.ort.edu.uy

Ejercicio 5
Implementar un algoritmo que sume los elementos de una matriz en forma recursiva

Firma a utilizar; public int sumar(int fila, int col, int orden, int[][] m)

Donde fila es el valor de la máxima fila
Columna es el valor de la máxima columna
Orden es la dimensión de la matriz

Ejemplo

Sumar(2,2,2,m) = 45

2	**4**	**7**
1	**3**	**5**
6	**9**	**8**

Solución

```
public  int sumar(int fila, int col, int orden, int[][] m) {
    if (fila == 0 && col == 0) {
        return m[0][0];
    } else if (col < 0) {
        return sumar(fila - 1, orden, orden, m);
    } else {
        return sumar(fila, col - 1, orden, m) + m[fila][col];
    }
 }
```