ELO320
Primer Certamen 18/04/2001

1.- Codifique en Lenguage C el algoritmo Quicksort y su función principal Partition vistos en clase. Suponga datos enteros.

```
int Partition(int [], int , int );
void Quicksort(int A[], int p, int r) {
        int q;
        if (p < r) {
                q=Partition(A, p, r);
                Quicksort(A, p, q);
                Quicksort(A, q+1, r);
        }
}
```

```
int Partition (int A[], int p, int r) {
int x = A[p], int i = p-1, int j = r+1, temp;
while (1) {
        while (A[--j] > x);
        while (A[++i] < x);
        if ( i<j ) {
                temp=A[i];
                A[i]=A[j];
                A[j]=temp;
        } else
                return j;
}
```

2.-Se tiene un archivo orden.c que contiene un programa para ordenar números usando quicksort. Por alguna razón el desarrollador de la aplicación decidió almacenar Quicksort en el archivo quicksort.c y Partition en partition.c. Además creó un archivo orden.h que contiene algunas constantes usadas en orden.c. Escriba el archivo makefile para compilar esta aplicación.

```
orden: orden.o quicksort.o partition.o
        cc -o orden orden.o quicksort.o partition.o
orden.o: orden.c orden.h
        cc -c orden.c
quicksort.o: quicksort.c
        cc -c quicksort.c
partition.o: particion.c
        cc -c partition.c
```

3.- Sean a=3, b=0x10 (hexadecimal), y c=013 (octal) enteros. Para cada segmento indique el valor de resultado (res). En todos los casos los valores inicial es son los mismos ya indicados.
a=3, b=16 c=11

Expresión	Valor de res	Expresión	Valor de res
temp = **c** - a++; res= temp & a;	**tem=8, a=4** **res = 0**	res = c>>3==1 && b!=10;	**Res=TRUE =1**
res = ~b;	**Res=0xFFFE =-17**	res = --a << 3;	**Res=16**
res = c^b;	**Res=27**	res = c &= b;	**Res=0**
res = c % a;	**Res=2**	res = b|c !=0x5?b/a:-b;	**Res=5**

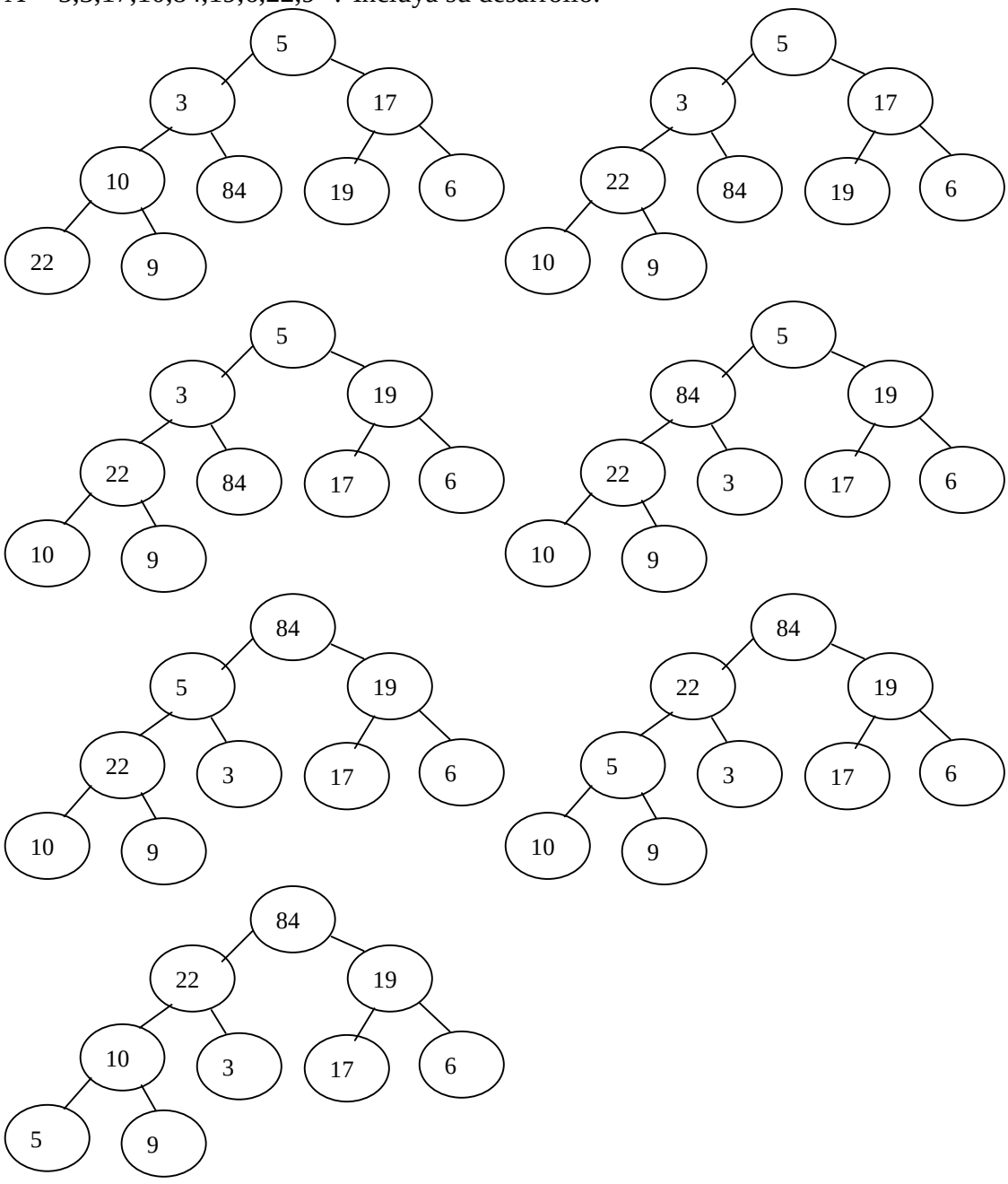
4.-El tiempo de ejecución del algoritmo de su empresa está descrito por la recurrencia $T(n)=\boldsymbol{a}T(n/4)+n^2$. El algoritmo de la competencia tiene un tiempo de ejecución $T'(n)=7T'(n/2)+n^2$. ¿Cuál es el mayor valor entero para ***a*** tal que su empresa tenga un algoritmo asintóticamente tan rápido como el de la competencia?

Análisis del algoritmo de la competencia

b=2, a=7 => $\log_b a = \log_2 7$, como $\log_2 7 > 2$ => estamos en el caso 1 del teorema maestro.

Luego T´(n) = $\Theta(n^{\log_2 7})$.
Análisis de mi algoritmo:
b=4, a= *a* para ser más rápido se debe cumplir $\log_4 a < \log_2 7 \Rightarrow a < 49 \Rightarrow a_{máx} = 49$.

5.- ¿Cuál es el resultado de aplicar la operación Build-heap sobre el arreglo A=<5,3,17,10,84,19,6,22,9>? Incluya su desarrollo.



Resultado: A=<84, 22, 19, 10, 3, 17, 6, 5, 9>

6.-Haga un árbol binario en que se muestre cada una de las particiones generadas por Quicksort al ser aplicado al arreglo A=<13,19,9,5,12,8,7,4,11,2,6,21>.

<13,19,9,5,12,8,7,4,11,2,6,21>

<6,2,9,5,12,8,7,4,11> <19,13,21>
<4,2,5> <9,12,8,7,6,11> <13> <19,21>
<2> <4,5> <6,7,8> <12,9,11> <13> <19> <21>
<2> <4> <5> <6> <7,8> <11,9> <12> <13> <19> <21>
<2> <4> <5> <6> <7> <8> <9> <11> <12> <13> <19> <21>

7.- El siguiente programa determina el máximo valor en un arreglo no ordenado A[1..n].

```
max = -"infinito";
for (i=1; i <=n; i++) {
        if (A[i] > max)
                max = A[i];  /* linea 4 */
```

Se desea determinar el número promedio de veces que la asignación de la línea 4 es ejecutada. Suponga que los números en A fueron tomados aleatoriamente del intervalo [0,1].

a) Si un número x es aleatoriamente escogido dentro de un conjunto de n número distintos, ¿cuál es la probabilidad que x sea el mayor número del conjunto?

1/n

b) Cuando la línea 4 es ejecutada, ¿cuál es la relación entre A[i] y A[j] para 1 <=j <= i?

A[i] > A[j]

c) Para cada i en el rango 1 <= i <=n, ¿cuál es la probabilidad que la línea 4 sea ejecutada?

1/i

d) Sea $s_1,s_2,..,s_n$ variables aleatorias, donde si representa el número de veces (0 ó 1) que la línea 4 es ejecutada durante la i-ésima iteración. ¿Cuál es el valor esperado $E[s_i]$?

$E[s_i]$ = 0*P(que línea 4 no sea ejecutada) + 1*P(línea 4 sí sea ejecutada) = 1/i

e) Determine el valor esperado para el número total de veces que la línea 4 es ejecutada.

E[número de veces pedido] = $E[s_1+s_2+s_3+...+s_n] = E[s_1]+E[s_2]+...+E[s_n]=1+1/2+1/3+...+1/n$
Como nota interesante (fuera del certamen) $1+1/2+1/3+...+1/n = \Theta(\lg n)$.