

Lista 9 - Tutoria

1. 1 fase:

	VETEBEBETES									
		Prox	Raiz	Folha						
a)	<table><tr><td>5</td><td>2</td><td>2</td><td>1</td><td>1</td></tr></table>	5	2	2	1	1	5	5	5	
5	2	2	1	1						
b)	<table><tr><td>5</td><td>2</td><td>2</td><td>1</td><td>2</td></tr></table>	5	2	2	1	2	5	5	3	
5	2	2	1	2						
c)	<table><tr><td>5</td><td>2</td><td>2</td><td>4</td><td>4</td></tr></table>	5	2	2	4	4	4	4	2	
5	2	2	4	4						
d)	<table><tr><td>5</td><td>2</td><td>2</td><td>4</td><td>4</td></tr></table>	5	2	2	4	4	3	4	2	
5	2	2	4	4						
e)	<table><tr><td>5</td><td>2</td><td>6</td><td>3</td><td>4</td></tr></table>	5	2	6	3	4	2	3	1	
5	2	6	3	4						
f)	<table><tr><td>2</td><td>7</td><td>2</td><td>3</td><td>4</td></tr></table>	2	7	2	3	4	1	0	0	
2	7	2	3	4						

```

ProximaFolha (A, n)
{
  Raiz = n; Folha = n;
  for (Prox = n; n >= 2; Prox--)
  {
    /* Procura Proxima x!
    if ((!no existe Folha) || ((Raiz > Prox) && (A[Raiz] <= A[Folha])))
    {
      A[Prox] = A[Raiz]; A[Raiz] = Prox; Raiz = Raiz - 1; /* No interno x!
    }
    else { A[Prox] = A[Folha]; Folha = Folha - 1; /* No folha x!
    }
    /* Atualize Frequencias x!
    if ((!no existe Folha) || ((Raiz > Prox) && (A[Raiz] <= A[Folha])))
    {
      /* No interno x!
      A[Prox] = A[Prox] + A[Raiz]; A[Raiz] = Prox; Raiz = Raiz - 1;
    }
    else { A[Prox] = A[Prox] + A[Folha]; Folha = Folha - 1; /* No folha x!
    }
  }
}

```

```

PrimeiraFase (A, n)
{ Raiz = n; Folha = 0;
  for (Prox = n; n >= 2; Prox--)
  { /* Procura Posicao */
    if ((! (nao existe Folha)) || ((Raiz > Prox) && (A[Raiz] <= A[Folha])))
    { A[Prox] = A[Raiz]; A[Raiz] = Prox; Raiz = Raiz - 1; /* No interno */ }
    else { A[Prox] = A[Folha]; Folha = Folha + 1; /* No folha */ }
    /* Atualiza Frequencias */
    if ((! (nao existe Folha)) || ((Raiz > Prox) && (A[Raiz] <= A[Folha])))
    { /* No interno */
      A[Prox] = A[Prox] + A[Raiz]; A[Raiz] = Prox; Raiz = Raiz - 1;
    }
    else { A[Prox] = A[Prox] + A[Folha]; Folha = Folha + 1; /* No folha */ }
  }
}

```

2 fase:

```

SegundaFase (A, n)
{ A[2] = 0;
  for (Prox = 3; Prox <= n; Prox++) A[Prox] = A[A[Prox]] + 1;
}

```

g)

2	0	1	2	3
---	---	---	---	---

3 fase:

Disp armazena quantos nós estão disponíveis no nível **h** da árvore.
u indica quantos nós do nível **h** são internos.

```
TerceiraFase (A, n)
{ Disp = 1; u = 0; h = 0; Raiz = 2; Prox = 1;
  while (Disp > 0)
  { while (Raiz <= n && A[Raiz] == h) { u = u + 1; Raiz = Raiz + 1; }
    while (Disp > u) { A[Prox] = h; Prox = Prox + 1; Disp = Disp - 1; }
    Disp = 2 * u; h = h + 1; u = 0;
  }
}
```

Disp: 1 2 1 2 1 2 1 2 1 0 0
u: 0 1 0 1 0 1 0 1 0 0
h: 0 1 2 3 4 5
raiz: 2 3 4 5 6
Prox: 1 2 3 4 5 6

h)

1	2	3	4	4
---	---	---	---	---