**Adiar o estouro de L e R no MPrim**

1 – Com alpha, delimitar a faixa de vértices que podem ser escolhidos;

2 – Dentro dessa faixa, separar os vértices em:

ta – aqueles que são terminais;

tb – aqueles que não alteram a contagem de L e de R, ou seja, aqueles que já são R;

tc – aqueles que diminuem o número de L e aumentam o número de R, aqueles que já são elos;

td – aqueles que aumentam o número de L apenas, aqueles que são folhas e não são terminais;

3 – Após a separação, escolher o próximo vértice obedecendo aos seguintes critérios:

* Se houverem vértices ta, escolher aleatoriamente um deles; senão:
* Se houverem vértices tb, escolher aleatoriamente um deles; senão:
* Se R estiver mais próximo de RMax do que L de LMax:
  + Se houverem vértices tc, escolher aleatoriamente um deles; senão:
* Se L estiver mais próximo de LMax do que R de RMax:
  + Se houverem vértices td, escolher aleatoriamente um deles; senão:
* Escolher aleatoriamente um vértice dentro da faixa alpha.

Para cada tipo de vértice, criar uma lista.

Fazer um teste escolhendo o menor da lista e outro escolhendo aleatoriamente na lista.

Para cada vértice que não for terminal na faixa:

Se o grau for 2, é link;

Se for maior que 2, é router.

Manter um vetor degree que é atualizado a cada iteração com os graus de cada vértice os graus serão através de inMST + parentVec e serão recontados a cada iteração.