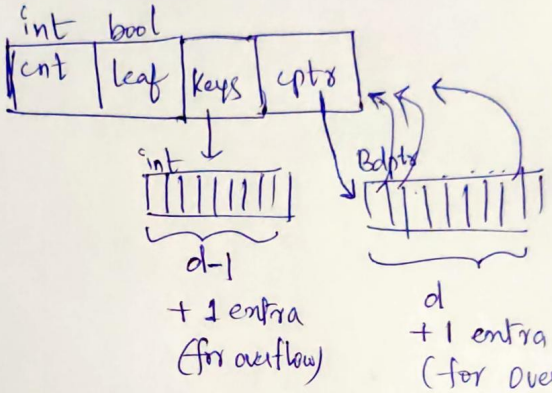
(A) B-3

Structure and diagram:-

```
struct Bdnode
{ int cnt=0;
  int *keys;
  bdptr *cptr;
  bool leaf;
};
```



Algorithm:-

- Left biased, and right biased occur during splitting of nodes during overflow and when d is even.
- when d is even say d=4, then during overflow we will have d+1=5 child pointers and d=4 keys.

→ for getting left biased, middle index should be $d/2$ such that left node will have more num of keys than right.

→ for getting right biased, middle index should be $\frac{d}{2}-1$, so that right node will have more num of keys than left.

→ Hence, I choose middle index according to the biasing, in the function void split();