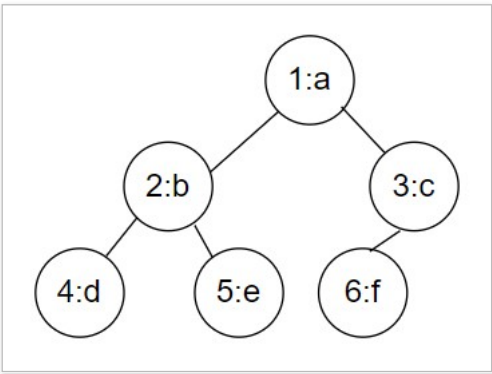


# 1.5 完全二叉树递归创建代码实战\_物联网 / 嵌入式工程师 - 慕课网

“ 慕课网慕课教程 1.5 完全二叉树递归创建代码实战涵盖海量编程基础技术教程，以图文图表的形式，把晦涩难懂的编程专业用语，以通俗易懂的方式呈现给用户。

## 5. 完全二叉树递归创建代码实战



有 n 个结点, 对于编号为 i(i>=1) 的结点:

- (1) 左孩子存在:  $2 * i \leq n$ (节点的个数), 左孩子编号为  $2 * i$
- (2) 右孩子存在:  $2 * i + 1 \leq n$ , 右孩子编号  $2 * i + 1$

bitree.h

```
#ifndef __BITREE_H__
#define __BITREE_H__

#include <string.h>
#include <stdio.h>
#include <stdlib.h>

#define N 6

typedef char data_t;

typedef struct bitree
{
    int n;
    data_t data;
    struct bitree *lchild;
    struct bitree *rchild;
}bitree_t;

extern bitree_t *create_binatry_tree(int n);

#endif
```

bitree.c

```
#include "bitree.h"

bitree_t *create_binatry_tree(int n)
{
    bitree_t *root = NULL;
```

```
root = (bitree_t *)malloc(sizeof(bitree_t));
memset(root,0,sizeof(bitree_t));

root->n = n;
root->lchild = root->rchild = NULL;

printf("Input %d node data : ",n);
scanf("%c",&(root->data));

while(getchar() != '\n');

if(2 * n <= N)
{
    root->lchild = create_binatry_tree(2 * n);
}

if(2 * n + 1 <= N)
{
    root->rchild = create_binatry_tree(2 * n + 1);
}

return root;
}
```

main.c

```
#include "bitree.h"

int main()
{
    bitree_t *root;

    root = create_binatry_tree(5);

    printf("create is successful!\n");
    return 0;
}
```

运行结果:

```
Input 1 node data: A
Input 2 node data: B
Input 4 node data: D
Input 5 node data: E
Input 3 node data: C
Input 6 node data: F
create is successful!
```

---

全文完

本文由 简悦 SimpRead 优化，用以提升阅读体验

使用了 全新的简悦词法分析引擎 beta，点击查看详细说明

