```c
char* getPermutation(int n, int k) {
    int group_count;
    int group_digit;
    int group;
    int remain;
    int index;
    int last_index;
    int result_index;
    bool* map;
    char* result;


    group_digit = n;
    map = (bool*)calloc(n, sizeof(bool));
    result = (char*)malloc(sizeof(char)*(n+1));
    result_index = 0;
    last_index = 0;

    while(group_digit>0)
    {
        group_digit--;
        group_count = 1;

        for(index = 2; index <= group_digit; index++)
        {
            group_count *= index;
        }

        group = (k-1)/group_count;
        last_index = 0;
        index = 0;

        while(index < n)
        {
            if(map[index] == false)
            {
                if(last_index == group)
                {
                   break;
                }

                last_index++;
            }

            index++;
        }

        map[index] = true;
        result[result_index] = '0' + index + 1;
        result_index++;
        k = k - group * group_count;

    }

    result[result_index] = '\0';

    return result;

}
```