相当于二分查找得到升级版

将中间值mid替换了

公式如下

$$mid=low+(high-low)(x-nums[low])/(nums[high]/nums[low])$$

```c
/*************************************************************************
    > File Name: bisection.c
    > Author: Sakura7301
    > Email: sakuraduck@foxmail.com
    > Github: https://github.com/Sakura7301
    > Created Time: 2021年07月28日 星期三 19时10分07秒
 ************************************************************************/

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

void search(int nums[],int size,int x)
{
    int left=0;
    int right=size-1;
    int flag=0;
    int sum=1;
    while(left<=right)
    {
        int mid;
        //mid=(left+right)/2;
        mid=left+(right-left)*(x-nums[left])/(nums[right]-nums[left]);
        printf("%dth\t:left=%d\tmid=%d\tright=%d\n",sum++,nums[left],nums[mid],nums[right]);
        if(x==nums[mid])
        {
            printf("第%d次查找到值为%d的数\n",sum-1,x);
            flag=1;
            break;
        }
        else if(x<nums[mid])
        {
            right=mid-1;
        }
        else if(x>nums[mid])
        {
            left=mid+1;
        }
    }
    if(flag==0)
    {
        printf("没有找到相关值。\n");
    }
}
int main()
{
    int x;
    int nums[]={0,7,12,13,19,31,40,55,66,77,88};
    int size=sizeof(nums)/sizeof(nums[0]);
    printf("Input your number:\n");
```

```
    scanf("%d",&x);
    search(nums,size,x);
    return 0;
}
```