

# 装载问题 - 分支限界算法

作者: [Saber](#)

原文链接: <https://ld246.com/article/1517112568414>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

# 装载问题

集装箱装载问题要求在不超过轮船载重量的前提下，将尽可能多的集装箱装上船；

## 样例：

输入：

80 4

18 7 25 36

输出：

79

## 分析：

本题可采用FIFO队列式分支限界算法，将解空间树的所有节点按照广度搜索的顺序排列成一个先进先出的队列，并用-1作为每一层的分割符。

## C++实现：

```
#include "iostream"  
#include "queue"  
  
#define NUM 100  
  
using namespace std;  
  
  
int n;  
int c;  
int w[NUM];
```

```
int MaxLoad()
{
    queue<int> Q;
    Q.push(-1);
    int i=0;
    int Ew=0;
    int bestw=0;
    int r=0;
    for(int j=1;j<n;j++)
    {
        r+=w[j];
    }
    while(1)
    {
        int wt=Ew+w[i];
        if(wt<=c)
        {
            if(wt>bestw)
                bestw=wt;
            if(i<n-1)
                Q.push(wt);
        }
        if(Ew+r>bestw&&i<n-1)
            Q.push(Ew);
        Ew=Q.front();
        Q.pop();
        if(Ew== -1)
```

```

    {
        if(Q.empty())
            return bestw;
        Q.push(-1);
        Ew=Q.front();
        Q.pop();
        i++;
        r-=w[i];
    }
}

return bestw;
}

int _tmain(int argc, _TCHAR* argv[])
{
    int x;
    cout<<"请输入轮船载重量和集装箱个数: ";
    cin>>c>>n;
    cout<<"每个集装箱的重量为: ";
    for(int i=0;i<n;i++)
    {
        cin>>x;
        w[i]=x;
    }
    cout<<"最大装载量为: "<<MaxLoad()<<endl;
}

return 0;

```

}