第二次算法作业——背包问题

09019106 牟倪

目录

[0-1背包问题 2](#_Toc68107419)

[有个数限制的0-1背包问题 3](#_Toc68107420)

[必须装满的0-1背包问题 4](#_Toc68107421)

0-1背包问题

设有n个物品，每一个物品的体积为ci、价值为wi。求用容量为C的背包最多能装多少价值的物品。输入：n、C；接下来n行，每一行为ci、wi。输出：答案。

1. #include<iostream>
2. #include<cstring>
3. using namespace std;
4. int n,C;int w[100000],c[100000],dp[100000];
5. int main(){
6. ios::sync\_with\_stdio(false);
7. memset(dp,0,sizeof(dp));
8. cin>>n>>C;
9. for(int i=1;i<=n;++i)
10. cin>>c[i]>>w[i];  *// 从1开始计数*
11. for(int i=1;i<=n;++i)
12. for(int s=C;s>0;--s)
13. if(s>=c[i]&&dp[s]<dp[s-c[i]]+w[i])
14. dp[s]=dp[s-c[i]]+w[i];
15. cout<<dp[C];
16. return 0;
17. }

测试数据：

输入：

5 100

77 92

22 22

29 87

50 46

99 90

输出：

133（取第3、4个）

有个数限制的0-1背包问题

限制物品的个数不能大于N。输入：n、C、N；接下来n行，每一行为ci、wi。输出：答案。

1. #include<iostream>
2. #include<cstring>
3. using namespace std;
4. int n,C,N;int w[100000],c[100000],dp[100000][100];
5. int main(){
6. ios::sync\_with\_stdio(false);
7. memset(dp,0,sizeof(dp));
8. cin>>n>>C>>N;
9. for(int i=1;i<=n;++i)
10. cin>>c[i]>>w[i];  *// 从1开始计数*
11. for(int i=1;i<=n;++i)
12. for(int s=C;s>0;--s)
13. for(int j=N;j>0;--j)
14. if(s>=c[i]&&dp[s][j]<dp[s-c[i]][j-1]+w[i])
15. dp[s][j]=dp[s-c[i]][j-1]+w[i];
16. cout<<dp[C][N];
17. return 0;
18. }

测试数据：

输入：

5 150 3

77 92

22 22

29 87

50 46

99 90

输出：

201（取第1、2、3个）

必须装满的0-1背包问题

背包必须装满，物品个数无限制。输入：n、C；接下来n行，每一行为ci、wi。输出：答案。

1. #include<iostream>
2. #include<cstring>
3. using namespace std;
4. int n,C;int w[100000],c[100000],dp[100000];
5. int main(){
6. ios::sync\_with\_stdio(false);
7. memset(dp,0,sizeof(dp));
8. cin>>n>>C;
9. for(int i=1;i<=n;++i)
10. cin>>c[i]>>w[i];  *// 从1开始计数*
11. for(int i=1;i<=n;++i)
12. for(int s=C;s>0;--s)
13. if(s>=c[i]&&(s-c[i]==0||dp[s-c[i]]!=0)&&dp[s]<dp[s-c[i]]+w[i])
14. dp[s]=dp[s-c[i]]+w[i];
15. cout<<dp[C];
16. return 0;
17. }

另解：

1. #include<iostream>
2. #include<cstring>
3. using namespace std;
4. const int inf=5e8;
5. int n,C;int w[100000],c[100000],dp[100000];
6. int main(){
7. ios::sync\_with\_stdio(false);
8. for(int s=0;s<100000;++s)dp[s]=-inf;dp[0]=0;
9. cin>>n>>C;
10. for(int i=1;i<=n;++i)
11. cin>>c[i]>>w[i];  *// 从1开始计数*
12. for(int i=1;i<=n;++i)
13. for(int s=C;s>0;--s)
14. if(s>=c[i]&&dp[s]<dp[s-c[i]]+w[i])
15. dp[s]=dp[s-c[i]]+w[i];
16. cout<<dp[C];
17. return 0;
18. }

测试数据：

输入：

5 100

50 92

50 22

49 87

19 46

30 90

输出：

114（取第1、2个）