**贪心算法实验活动安排问题**

**代码：**

#include <stdio.h>

int ActiveManage(int s[100], int f[100], bool a[100], int n)

{

a[1] = 1;

int j = 1, count = 1;

for (int i = 2; i <= n; i++)

{

if (s[i] >= f[j])

{

a[i] = 1;

j = i;

count++;

}

else

a[i] = 0;

}

return count;

}

int main()

{

int n;

int s[100], f[100];

bool a[100] = { 0 };

printf("输入活动的个数：\n");

scanf("%d", &n);

printf("输入各个活动的开始和结束时间：\n");

for (int j = 1; j <= n; j++)

scanf("%d%d", &s[j], &f[j]);

int ans = ActiveManage(s, f, a, n);

printf("最多安排%d个活动，分别是：\n", ans);

for (int k = 1; k <= n; k++)

{

if (a[k] == 1)

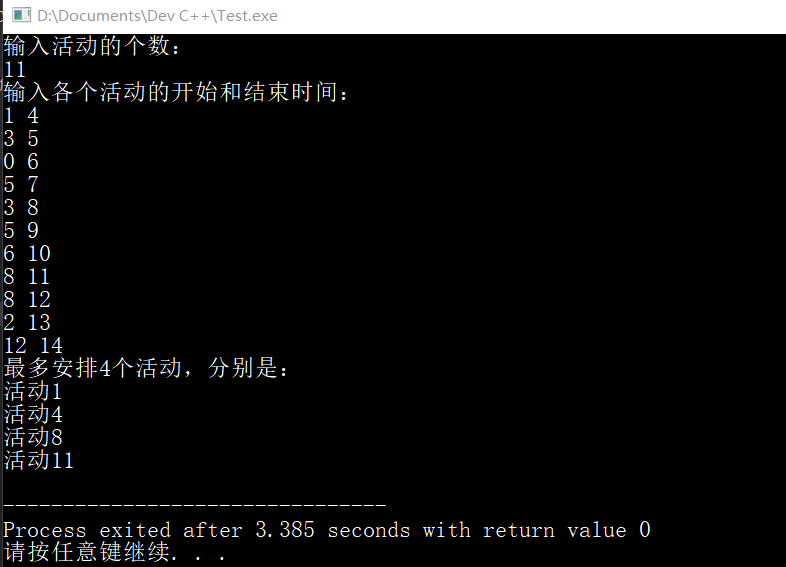
printf("活动%d\n", k);

}

return 0;

}

**结果：**



**背包问题**

**代码：**

#include<stdio.h>

# define MAX 100

int min(int a, int b)

{

if (a < b)

{

return a;

}

else

{

return b;

}

}

int max(int a, int b)

{

if (a > b)

{

return a;

}

else

{

return b;

}

}

void kna(int n, int c, int v[MAX], int w[MAX], int m[MAX][MAX])

{

int jmax = min(w[n - 1] - 1, c);

for (int j = 0; j <= jmax; j++)

{

m[n][j] = 0;

}

for (int j = w[n - 1]; j <= c; j++)

{

m[n][j] = v[n - 1];

}

for (int i = n - 1; i > 1; i--)

{

jmax = min(w[n - 1] - 1, c);

for (int j = 0; j <= jmax; j++)

{

m[i][j] = m[i + 1][j];

}

for (int j = w[i]; j <= c; j++)

{

m[i][j] = max(m[i + 1][j], m[i + 1][j - w[i]] + v[i]);

}

}

m[1][c] = m[2][c];

if (c >= w[1])

{

m[1][c] = max(m[1][c], m[2][c - w[1]] + v[1]);

}

}

void tra(int m[MAX][MAX], int w[MAX], int c, int n, int x[MAX])

{

for (int i = 0; i < n; i++)

if (m[i][c] == m[i + 1][c])

{

x[i] = 0;

}

else

{

x[i] = 1;

c = c - w[i];

}

x[n - 1] = (m[n][c]) ? 1 : 0;

}

int main()

{

int n, c;

scanf("%d", &n);

scanf("%d", &c);

int v[MAX] = { 0 }, w[MAX] = { 0 }, x[MAX] = { 0 };

for (int i = 0; i < n; i++)

{

scanf("%d", &w[i]);

scanf("%d", &v[i]);

}

int m[MAX][MAX] = { 0 };

kna(n, c, v, w, m);

tra(m, w, c, n, x);

printf("%d\n", m[1][c]);

for (int i = 0; i < n; i++)

{

printf("%d ", x[i]);

}

return 0;

}

**结果：**

