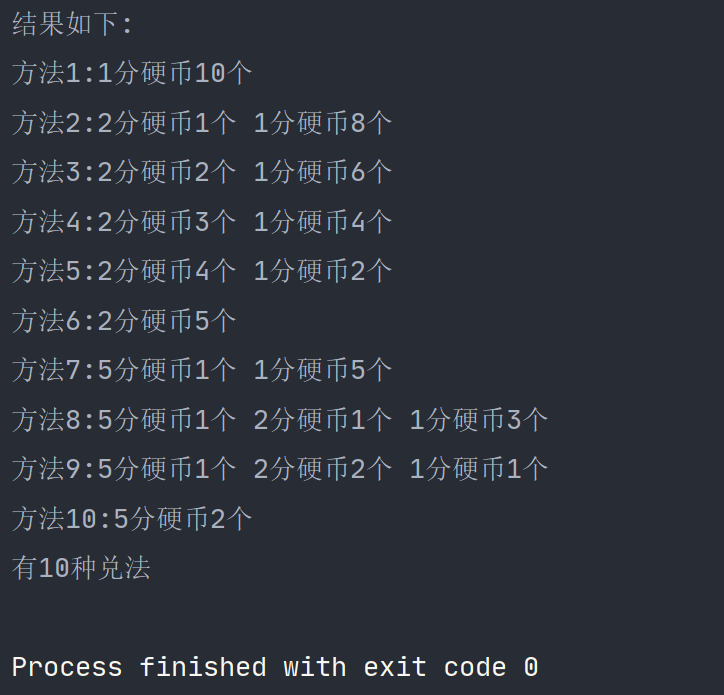
### P160 实验四

### 实验题2

代码：

*package* test2;  
  
*public class* test2 {  
 *public static void* main(String[] args) {  
 *int* n = 10;  
 *change*(n);  
 }  
  
 *public static void* change(*int* n) {  
 *int* x = 0;  
 *int* y = 0;  
 *int* z = 0;  
 *int* count = 0;  
 *for* (z = 0; z <= n / 5; z++) {  
 *for* (y = 0; y <= (n - z \* 5) / 2; y++) {  
 *if* (5 \* z + 2 \* y <= n) {  
 x = n - 5 \* z - 2 \* y;  
 System.out.printf("方法%d:", ++count);  
 *if* (z != 0) {  
 System.out.printf("5分硬币%d个 ", z);  
 }  
 *if* (y != 0) {  
 System.out.printf("2分硬币%d个 ", y);  
 }  
 *if* (x != 0) {  
 System.out.printf("1分硬币%d个", x);  
 }  
 System.out.println();  
 }  
 }  
 }  
 System.out.println("有" + count + "种兑法");  
 }  
}

结果：



### 实验题4

解：

分析：选择湖1—湖i中鱼最多的湖开始,假设最多的湖是湖k,在此湖钓鱼,每过一个单位时间,Lake[i].max+=cfi[k](增加的钓鱼数)，Lake[i].num[k]+=5(增加的钓鱼时间)，同时置cfi[k]-=di[k](湖k的鱼数量减少)。下一个单位时间重复进行,直到restT=0(每经过一个单位时间cfi会发生改变)。最后枚举每个湖i作为最后的湖,求出Lake数组,在其中找出max最大的湖maxlast即为所求。故该题目属于排列数问题。

代码：

#include<stdio.h>

#define MAX 30

int n = 2;//湖的个数

int h = 1;//可用时间

int fi[MAX] = {0, 10, 1};//最初钓鱼量,数组下标0不用

int di[MAX] = {0, 2, 5};//单位时间鱼的减少量

int ti[MAX] = {0, 2};//湖i到湖i+1的时间

int cfi[MAX];//保存fi

struct NodeType {

int num[MAX];

int max;//最多的钓鱼量

} Lake[MAX]; //Lake[i]表示经过最后一个湖i的结果

int maxlast;//最多钓鱼量时最后经过的湖的编号

int GetMax(int p[], int i, int j) {//求最大元素的下标

int maxi = i;

for (int k = i + 1; k <= j; k++) {

if (p[maxi] < p[k]) {

maxi = k;

}

}

return maxi;

}

void solve() {

int i, j, t, restT;

int T = 60 \* h;

for (i = 1; i <= n; i++) {

restT = T;

for (j = 1; j <= i; j++) {

cfi[j] = fi[j];

if (j < i) {

restT -= 5 \* ti[j];

}

}

t = 0;

while (t < restT) {

int k = GetMax(cfi, 1, i);

Lake[i].max += cfi[k];

Lake[i].num[k] += 5;

if (cfi[k] >= di[k]) {

cfi[k] -= di[k];

}

else {

cfi[k] = 0;

}

t += 5;

}

}

}

int main() {

int i, j;

for (i = 1; i <= n; i ++) {

Lake[i].max = 0;

for (j = 0; j <= n; j ++)

Lake[i].num[j] = 0;

}

solve();

maxlast = 1;

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

if (Lake[i].max > Lake[maxlast].max) {

maxlast = i;

}

}

for (i = 1; i <= n; i++) {

printf("在湖%d钓鱼时间为%d分钟\n", i, Lake[maxlast].num[i]);

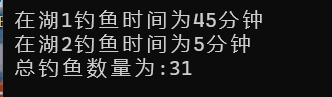
}

printf("总钓鱼数量为:%d\n", Lake[maxlast].max);

return 0;

}

结果：



### 在线编程题1

代码：

*package* test1;  
  
*import* java.util.Scanner;  
  
*public class* test1 {  
 *public static void* main(String[] args) {  
 Scanner scanner = *new* Scanner(System.in);  
 *long* a = scanner.nextLong();  
 *long* b = scanner.nextLong();  
 *long* c = scanner.nextLong();  
 *long* d = scanner.nextLong();  
 *solve*(a, b, c, d);  
 }  
  
 *public static void* solve(*long* a, *long* b, *long* c, *long* d) {  
 *double* x = 0;  
 *for* (*int* i = -10000; i <= 10000; i++) {  
 x = i / 100.0;  
 *double* f = a \* x \* x \* x + b \* x \* x + c \* x + d;  
 *if* (f > -0.0001 && f < 0.0001) {  
 System.out.printf("%.2f ", x);  
 }  
 }  
 System.out.println();  
 }  
}

结果：

