

# 2016-2017 学年第一学期期末考试试题(B)卷

## 参考答案和评分标准

课程名称 《算法设计与分析》 任课教师签名\_\_\_\_\_

出题教师签名 题库抽题 审题教师签名\_\_\_\_\_

考试方式 (闭) 卷 适用专业 2014 级计算机科学与技术

考试时间 ( 120 ) 分钟

### 一、单项选择 (每小题 2 分, 共 30 分)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
D	C	B	A	A	D	C	A	D	C	D	A	C	A	D

评分标准: 答对 1 题得 2 分, 答错不得分。

### 二、填空 (每空 1 分, 共 10 分)

- 1、问题分析、算法分析
- 2、事后分析法
- 3、算法程序本身所占空间
- 4、数据项、数据元素、数据元素的关系
- 5、迭代次数可知、迭代次数无法确定
- 6、回溯

评分标准: 答对 1 空得 1 分, 答错不得分。

### 三、算法应用 (每题 10 分, 共 40 分)

#### 1、算法策略: 贪心法

先尽可能多地找大面额, 面额需要递减有序。

37889 元的方案: 378 张 100 元, 1 张 50 元, 1 张 20 元, 1 张 10 元, 1 张 5 元, 2 张 2 元。

评分标准:

答对算法策略得 2 分, 答错不得分;  
算法步骤答对得 5 分, 答错不得分;  
零钱方案答对每个得 0.5 分, 答错不得分。

#### 2、算法策略: 迭代法或者枚举法。

迭代法步骤如下:

- 1) 先利用辗转相除求出  $a, b$  两数的最小公倍数  $lcm$ ;
- 2) 再求出  $lcm$  和  $c$  的最小公倍数。  
81, 63 和 91 的最小公倍数为 7371。

评分标准:

答对算法策略得 2 分, 答错不得分;  
算法步骤答对得 5 分, 答错不得分;  
最小公倍数答对得 3 分, 答错不得分。

#### 3、算法策略: 分治法或者递推法 (动态规划)。

递推法步骤如下:

下标从 1 到  $n$  循环

如果当前和大于 0, 则加上现下标元素;

否则重新开始, 并标注当前起点;

如果有更大的当前和, 标注下最大和与序列起点。

最大子段和为 89, 区间为(7, 41, 28, -29, 15, 27)。

评分标准:

答对算法策略得 2 分, 答错不得分;  
算法步骤答对得 5 分, 答错不得分;  
最大子段和答对得 2 分, 答错不得分;  
最大子段和区间答对得 1 分, 答错不得分。

#### 4、算法策略: 贪心法。

将初始序列递增和递减排序得到: 2, 3, 4, 5, 6, 7 和 7, 6, 5, 4, 3, 2, 分别取出序列中最小值和最大值依次各自计算得到:

步骤	$max$	$min$
初始	2, 3, 4, 5, 6, 7	7, 6, 5, 4, 3, 2
第 1 步	$2 + 3 + 1 = 7$ , 4, 5, 6, 7	$7 + 6 + 1 = 43$ , 5, 4, 3, 2
第 2 步	$4 + 5 + 1 = 21$ , 6, 7, 7	$43 + 5 + 1 = 216$ , 4, 3, 2
第 3 步	$6 + 7 + 1 = 43$ , 7, 21	$216 + 4 + 1 = 865$ , 3, 2
第 4 步	$7 + 21 + 1 = 148$ , 43	$865 + 3 + 1 = 2596$ , 2
第 5 步	$43 + 148 + 1 = 6365$	$2596 + 2 + 1 = 5193$
结果	极差 $= max - min = 1172$	

评分标准:

答对算法策略得 2 分, 答错不得分;

求极差的步骤答对得 2.5 分, 答错不得分;

极差计算步骤正确每 1 步得 0.5 分, 计算错误不得分;

级差结果计算正确得 3 分, 错误不得分。

#### 四、算法设计 (每题 10 分, 共 20 分)

1、

```
#include <stdio.h>
int main( )
{
    int a, i;
    a = 1;
    for (i = 9; i >= 1; i --)
        a = (a + 1) * 2;
    printf("%d\n", a);
    return 0;
}
```

评分标准:

算法正确得 3 分, 错误不得分;

使用倒推法的算法策略得 5 分, 答错不得分;

正确输出得 2 分, 错误不得分。

2、

```
#include <stdio.h>
int f(int n)
{
    if (n == 1)
        return 1;
    if (n == 2)
        return 2;
    return f(n - 1) + f(n - 2);
}
int main( )
{
```

```
int n;
scanf("%d", &n);
printf("%d\n", f(n));
return 0;
}
```

评分标准:

算法正确得 3 分, 错误不得分;

使用递归或者递推的算法策略得 5 分, 答错不得分;

正确输入输出得 2 分, 错误不得分。