

开始时间	2019年05月14日 星期二 10:16
状态	完成
完成于	2019年05月14日 星期二 11:21
耗时	1 小时 4 分钟
成绩	100.00/满分100.00

题目1

正确

获得1.50分中的1.50分

Which application can be developed based on the graph data structure?

选择一项:

- ☐ Student management system
- ☐ File directory
- ☐ Family information system
- ☒ Transportation networks ✓

你的回答正确

题目2

正确

获得1.50分中的1.50分

在用贪心策略设计算法时，关键时（ ）的确定和证明。

选择一项:

- ☒ 贪心准则 ✓
- ☐ 动态规划方程
- ☐ 合并解的方法
- ☐ 分解子问题的方法

你的回答正确

题目3

正确

获得1.50分中的1.50分

下列程序段（算法）的时间复杂度为（ ）。

```
x=n;
y=0;
while(x>=(y+1)*(y+1))
    y=y+1;
```

选择一项:

- ☒ $O(\sqrt{n})$ ✓
- ☐ $O(n)$
- ☐ $O(n^2)$
- ☐ $O(1)$

题目4

正确

获得1.50分中的1.50分

数据结构中，与计算机无关的是数据的（ ）结构。

选择一项:

- ☐ 物理
- ☐ 物理和存储
- ☒ 逻辑 ✓
- ☐ 存储

你的回答正确

题目5

正确

获得1.50分中的1.50分

算法分析的目的是（ ）。

选择一项:

- ☐ 找出数据结构的合理性
- ☒ 分析算法的效率以求改进 ✓
- ☐ 研究算法中输入和输出的关系
- ☐ 分析算法的易懂性和文档性

你的回答正确

题目6

正确

获得1.50分中的1.50分

计算机中的算法指的是解决某一个问题的有限操作序列，它必须具备具备输入、输出和（ ）等5个特性。

选择一项:

- ☐ 可行性、可移植性和可扩充性
- ☐ 确定性、有穷性和稳定性
- ☐ 易读性、稳定性和安全性
- ☒ 可行性、确定性和有穷性 ✓

你的回答正确

题目7

正确

获得1.50分中的
1.50分

数据结构采用顺序存储结构时，存储单元的地址（ ）。

选择一项：

- ☐ 不一定连续
- ☐ 部分连续，部分不连续
- ☐ 一定不连续
- ☒ 一定连续 ✓

你的回答正确

题目8

正确

获得1.50分中的
1.50分

以下数据结构中，（ ）是非线性数据结构。

选择一项：

- ☐ 队列
- ☐ 字符串
- ☐ 栈
- ☒ 树 ✓

你的回答正确

题目9

正确

获得1.50分中的
1.50分

Which one is not a part of an abstract data structure?

选择一项：

- ☒ Implementation of Operations ✓
- ☐ Data
- ☐ Abstract operations
- ☐ Structure(Relation)

你的回答正确

题目10

正确

获得1.50分中的
1.50分

一个算法必须总是（对任何合法的输入值）在执行有穷步之后结束，是指算法的（ ）特性。

选择一项：

- ☐ 确定性
- ☒ 有穷性 ✓
- ☐ 可行性
- ☐ 正确性

题目11

正确

获得1.50分中的
1.50分

What is the time cost of the following code:

```
// Find largest value
int Largest(int **array, int n) {
    int currlarge = array[0][0]; // Largest value seen
    for (int i=0; i<n; i++)
        for (int j = 0; j < n; j++)
            if (currlarge < array[i][j])
                currlarge = array[i][j];
    return currlarge;        // Return largest
}
```

选择一项：

- ☐ $T(n)=c1*i*j + c2$
- ☐ $T(n)=c1*n+c2$
- ☒ $T(n)=c1*n*n+c2$ ✓
- ☐ $T(n)=c1+c2$

你的回答正确

题目12

正确

获得1.50分中的
1.50分

Which one is slowest:

选择一项：

- ☐ $O(2^n)$
- ☐ $O(n^2)$
- ☒ $O(n!)$ ✓
- ☐ $O(\log(n))$

你的回答正确

题目13

正确

获得1.50分中的1.50分

O indicates the () bound.

选择一项:

- ☒ Upper ✓
- ☐ Mean
- ☐ Lower
- ☐ Exact

你的回答正确

题目14

正确

获得1.50分中的1.50分

Ω indicates the () bound.

选择一项:

- ☐ Exact
- ☐ Mean
- ☐ Upper
- ☒ Lower ✓

你的回答正确

题目15

正确

获得1.50分中的1.50分

计算机算法指的 (), 它必须具可读性、健壮性、高性能等特性。

选择一项:

- ☐ 排序方法
- ☐ 计算方法
- ☒ 解决问题的步骤序列 ✓
- ☐ 调度方法

你的回答正确

题目16

正确

获得1.50分中的1.50分

从 $n>=3$ 个互异整数中, 除最大、最小者以外, 任取一个“常规元素”。 算法如下:

```
ordinaryElement(S[ ],n)
{
    从S中任取三个元素x,y,z;
    通过比较, 对它们排序;
    输出中间元素; }

```

该算法的时间复杂性为 ()。

选择一项:

- ☐ $O(n)$
- ☒ $O(1)$ ✓
- ☐ $O(n\log_2(n))$
- ☐ $O(\log_2(n))$

题目17

正确

获得1.50分中的1.50分

存储密度是评价数据结构存储方案的一个指标。它定义为: 存储数据元素占用空间/分配给数据结构的总存储空间。

下面不正确的是 ()。

选择一项:

- ☒ 顺序存储方式比链式存储方式存储密度小 ✓
- ☐ 一般来说, 存储密度越大, 对数据的操作就越困难; 反之存储密度越小, 对数据的操作就越容易。
- ☐ 存储密度反映了存储空间的利用率
- ☐ 顺序存储方式比链式存储方式存储密度大

题目18

正确

获得1.50分中的1.50分

对于任意非负整数, 下面的算法统计其对应二进制中数位1的个数。

```
int countOnes(unsigned int n)
{
    int ones=0;
    while(n>0)
    {
        ones=ones + n%2; //%为取余运算
        n=n/2; }
    return ones;
}

```

该算法的时间复杂性为 ()。

选择一项:

- ☒ $O(\log_2(n))$ ✓
- ☐ $O(n)$
- ☐ $O(1)$
- ☐ $O(n^2)$

题目19

正确

获得1.50分中的1.50分

关于ADT, 下面不正确的描述是 ()。

选择一项:

- ☐ 一个ADT, 可以有不同的实现
- ☐ 两个ADT相同, 是指其数据结构和定义的操作分别都相同
- ☐ 对同一个问题, 不同人可以抽象定义不同的ADT
- ☒ 对一个问题抽象定义出的ADT只能有一种实现 ✓

<div>题目20</div> <div>正确</div> <div>获得1.50分中的1.50分</div>	<div>关于程序、数据结构和算法，下面正确的描述是（ ）。</div> <div>选择一项：</div> <div><div><input type="radio"/> 程序就是算法</div><div><input type="radio"/> 程序是由数据结构和算法组成的</div><div><input checked="" type="radio"/> 程序是对数据结构和算法的一种实现表示 ✓</div><div><input type="radio"/> 算法就是程序</div></div>
<div>题目21</div> <div>正确</div> <div>获得1.50分中的1.50分</div>	<div>关于抽象数据类型和数据类型，下面不正确的描述是（ ）。</div> <div>选择一项：</div> <div><div><input type="radio"/> 抽象数据类型本身与计算机无关</div><div><input checked="" type="radio"/> 抽象数据类型包括数据结构、存储结构和定义的操作 ✓</div><div><input type="radio"/> 一个数据类型的实现包括三个阶段：抽象定义、虚拟表示、物理实现</div><div><input type="radio"/> 高级语言提供的数据类型是一些物理实现了的抽象数据类型</div></div>
<div>题目22</div> <div>正确</div> <div>获得1.50分中的1.50分</div>	<div>关于渐近时间复杂性分析，不正确的是（ ）。</div> <div>选择一项：</div> <div><div><input type="radio"/> 它是一种事前估计算法时间复杂性的方法</div><div><input type="radio"/> 关注点是问题规模大到一定程度后算法的时间性能表现</div><div><input checked="" type="radio"/> 比较的是算法基本操作的次数 ✓</div><div><input type="radio"/> 比较的是算法基本操作次数随问题规模变化的增长率</div></div>
<div>题目23</div> <div>正确</div> <div>获得1.50分中的1.50分</div>	<div>关于设计算法的分治策略，下面错误的描述是（ ）。</div> <div>选择一项：</div> <div><div><input type="radio"/> 折半查找算法就是采用的分治策略，而且子问题的解就是原问题的解</div><div><input type="radio"/> 递归算法就是采用了分治策略</div><div><input type="radio"/> 分解和合并是关键</div><div><input checked="" type="radio"/> 简单的分解与合并就会产生好算法 ✓</div></div>
<div>题目24</div> <div>正确</div> <div>获得1.50分中的1.50分</div>	<div>关于数据结构，下面不正确的描述是（ ）。</div> <div>选择一项：</div> <div><div><input checked="" type="radio"/> 数据结构在计算机诞生之初就已经产生了 ✓</div><div><input type="radio"/> 数据结构本身与计算机无关</div><div><input type="radio"/> 求解简单问题，可以不用考虑数据结构</div><div><input type="radio"/> 随着计算机应用的深入，解决的问题越来越复杂。一方面发展硬件技术，另一方面通过研究分析问题的数据及特点，并利用其特性提高问题求解的效率，这就产生了数据结构</div></div>
<div>题目25</div> <div>正确</div> <div>获得1.50分中的1.50分</div>	<div>关于数据结构，正确的描述是（ ）。</div> <div>选择一项：</div> <div><div><input type="radio"/> 数据结构就是存储结构</div><div><input type="radio"/> 数据及其存储方式</div><div><input type="radio"/> 数据之间的逻辑关系</div><div><input checked="" type="radio"/> 数据及数据之间的逻辑关系 ✓</div></div>
<div>题目26</div> <div>正确</div> <div>获得1.50分中的1.50分</div>	<div>关于数据结构和存储结构的关系，错误的描述是（ ）。</div> <div>选择一项：</div> <div><div><input checked="" type="radio"/> 数据结构就是存储结构，存储结构就是数据结构 ✓</div><div><input type="radio"/> 一种存储结构可以存储不同的数据结构</div><div><input type="radio"/> 一种数据结构可以有多种不同的存储结构</div><div><input type="radio"/> 各种存储结构存储数据元素本身占用的存储空间大小是一样的，但是存储（表示）关系占用的空间不同。</div></div>
<div>题目27</div> <div>正确</div> <div>获得1.50分中的1.50分</div>	<div>关于算法，错误的描述是（ ）。</div> <div>选择一项：</div> <div><div><input type="radio"/> 算法必须有输出，但可以没有输入</div><div><input type="radio"/> 算法的每一步必须有严格定义和确定的动作</div><div><input checked="" type="radio"/> 程序就是算法，算法就是程序 ✓</div><div><input type="radio"/> 算法必须在执行有限步后能自动结束</div></div>
<div>题目28</div> <div>正确</div> <div>获得1.50分中的1.50分</div>	<div>关于算法的正确性，下面错误的描述是（ ）。</div> <div>选择一项：</div> <div><div><input type="radio"/> 运行算法，进行测试，不能保证算法的正确性</div><div><input type="radio"/> 算法的正确性应该通过数学证明手段来保证</div><div><input type="radio"/> 所谓正确性是指，对问题的任何输入，都能正确得到相应的输出</div><div><input checked="" type="radio"/> 运行算法，进行测试，可以保证算法的正确性 ✓</div></div>

题目29

正确

获得1.50分中的1.50分

链式存储结构除了存放数据元素外，还要存储关系，即存储与其相关元素的地址。假设数据元素是整数，每个元素只与一个元素有关系。下面能正确定义链式存储时元素存储结点的是（ ）。

选择一项：

- ☒ struct node { int x; struct node *p; }; ✓
- ☐ int x;
- ☐ struct node { int x; struct node p; };
- ☐ struct node { int x;

题目30

正确

获得1.50分中的1.50分

人机对弈问题的数据是棋局的状态，数据量非常大，这些数据之间具有（ ）关系。

选择一项：

- ☒ 层次关系 ✓
- ☐ 无
- ☐ 图关系
- ☐ 线性关系

题目31

正确

获得1.50分中的1.50分

如果一个算法的时间复杂度为O(n)，则下面不正确的是（ ）。

选择一项：

- ☐ 某些情况下，该算法可能比时间复杂度为O(n²)的算法性能差
- ☐ 该算法的最坏时间复杂度为线性增长率
- ☐ 如果计算机的运行速度提高K倍，那么在相同时间内，该算法可解决问题的规模也可以提高K倍
- ☒ 任何时候它比时间复杂度为O(n²)的算法性能都好 ✓

题目32

正确

获得1.50分中的1.50分

数据结构按其结构关系区分有三种，不包括（ ）。

选择一项：

- ☒ 集合 ✓
- ☐ 线性表
- ☐ 树
- ☐ 图

题目33

正确

获得1.50分中的1.50分

数据结构采用链式存储时，其存储空间地址（ ）。

选择一项：

- ☒ 不一定连续 ✓
- ☐ 部分连续
- ☐ 一定连续
- ☐ 一定不连续

题目34

正确

获得1.50分中的1.50分

顺序存储结构要求必须是地址连续的存储空间。假设有n个学生数据元素，每个学生包括学号、姓名、成绩等数据项。

```
struct node
{ int St_No;
  char St_Name[10];
  float St_Score;
};
```

下面能正确获取顺序存储空间的是（ ）。

选择一项：

- ☒ int n; struct node *St; cin>>n; St=(struct node *)malloc(sizeof(struct node)*n); ✓
- ☐ int n; struct node *St; cin>>n; St=(struct node *)malloc(n);
- ☐ int n; struct node *St; cin>>n; St=(struct node *)malloc(sizeof(struct node));
- ☐ int n; struct node *St; cin>>n; St = malloc(sizeof(struct node)*n);

题目35

正确

获得1.50分中的1.50分

算法的计算量的大小称为计算的（ ）。

选择一项：

- ☐ A. 现实性
- ☐ B. 效率
- ☐ C. 难度
- ☒ D. 复杂度 ✓

题目36

正确

获得1.50分中的1.50分

算法设计是一种创新行为，没有固定模式，但有一些经验总结。递归算法就是一种（ ）策略。

选择一项：

☐ 贪心

☐ 蛮力

☐ 动态规划

☒ 分治

题目37

正确

获得1.50分中的1.50分

下面是求a的n次幂的递归算法，其时间复杂度为（ ）。

```
int power(int a,int n)
{ int t;
  if(n==0) return 1;
  else if(n%2==0)
  { t=power(a,n/2);
    return t*t; }
  else { t=power(a,(n-1)/2);
        return a*t*t; }
}
```

选择一项：

☐ $O(n)$

☒ $O(\log_2(n))$

☐ $O(n^2)$

☐ $O(1)$

题目38

正确

获得1.50分中的1.50分

以下说法错误的是（ ）。

选择一项：

☐ 抽象数据类型具有封装性。

☐ 抽象数据类型的一个特点是使用与实现分离。

☐ 抽象数据类型具有信息隐蔽性。

☒ 使用抽象数据类型的用户可以自己定义对抽象数据类型中数据的各种操作。

题目39

正确

获得1.50分中的1.50分

有求n个数累加和的算法如下，其时间复杂度为（ ）。

```
int Sum(int a[ ],int n)
{ if(n==0) s=a[n];
  else s=a[n-1]+Sum(a,n-1);
  return s;
}
```

选择一项：

☐ $O(1)$

☐ $O(\log_2(n))$

☒ $O(n)$

☐ $O(n^2)$

题目40

正确

获得1.50分中的1.50分

有算法如下，其时间复杂度为（ ）。

```
x=0;
i=1;
while(i<=n)
{ x=x+i; i=i*2; } ;
```

选择一项：

☐ $O(1)$

☒ $O(\log_2(n))$

☐ $O(n)$

☐ $O(n^2)$

题目41
正确
获得20.00分中的
20.00分

分治策略是算法设计的重要策略之一，该策略的基本思想是把问题进行分解成一些问题，通过子问题的求解完成对原问题的求解。其关键是分解和合并，好的分解或合并方法才会产生高效的分治算法。

分治策略设计出的算法最常见的就是递归算法。但是如果在分解时，分解出的子问题有很多是重复的，那么这样的分治（递归）算法求解问题的效率就非常低。例如斐波那契数问题，如果采用递归求解，算法效率非常低： $O(2^n)$ 。而如果采用递推求解(动态规划自底向上求解)，算法效率非常高： $O(n)$ 。

现在请你编写程序，统计计算一个斐波那契数时分解出的各子问题的个数。

斐波那契数的定义如下：

Fib(0)=0
Fib(1)=1
Fib(n)=Fib(n-1)+Fib(n-2)

输入： 一个整数n，即计算Fib(n)

输出： n+1行，即各个子问题的值及该子问题的个数。

例如

输入：

5

输出：

Fib(0)=0,spn=3

Fib(1)=1,spn=5

Fib(2)=1,spn=3

Fib(3)=2,spn=2

Fib(4)=3,spn=1

Fib(5)=5,spn=1

答案： (penalty regime: 0 %)

```
1 #include <iostream>
2
3 using namespace std;
4
5
6 int a[1000];
7 int cnt[1000];
8
9 int Fib(int n)
10 {
11     if(n==0)
12     {
13         cnt[n]++;
14         return 0;
15     }
16     if(n==1)
17     {
18         cnt[n]++;
19         return 1;
20     }
```

	输入	期望输出	实际输出	
✓	1	Fib(0)=0, spn=0 Fib(1)=1, spn=1	Fib(0)=0, spn=0 Fib(1)=1, spn=1	✓
✓	5	Fib(0)=0, spn=3 Fib(1)=1, spn=5 Fib(2)=1, spn=3 Fib(3)=2, spn=2 Fib(4)=3, spn=1 Fib(5)=5, spn=1	Fib(0)=0, spn=3 Fib(1)=1, spn=5 Fib(2)=1, spn=3 Fib(3)=2, spn=2 Fib(4)=3, spn=1 Fib(5)=5, spn=1	✓

通过所有测试 ✓

正确
此次提交得分： 20.00/20.00。

题目42

正确

获得20.00分中的20.00分

假设有n个活动的集合E={a1,a2,...,an}，其每个活动都要求使用同一资源（如某个设备、教室、场地等），而在同一时间内只允许一个活动使用这一资源。

每个活动都有一个要求使用该资源的起止时间si,fi，且si<fi。如果选择了活动ai，则它在半开的时间区间[si,fi)内占有资源。两个活动ai,aj称为是相容的，当且仅当它们的时间区间[si,fi)和[sj,fj)不相交，即si>=fj 或 sj >=fi。现要求在所给定的活动集中选出最大的相容活动子集。

请补充要求的函数代码。

提示：贪心策略

输入，有多行，第1行是活动的个数n，后面n行，每行3个整数，是每个活动的编号、占用资源的开始时间、结束时间

输出，选出的最大活动子集，即有多行，每行包括活动的编号、开始时间、结束时间。

例如：

输入：

11
1 3 8
2 2 13
3 1 4
4 5 7
5 6 10
6 8 11
7 12 14
8 5 9
9 3 5
10 0 6
11 8 12
输出：
3: 1-4
4: 5-7
6: 8-11
7: 12-14

答案： (penalty regime: 0 %)

```
1 #include <iostream>
2 #include <stdio.h>
3 #include <stdlib.h>
4 using namespace std;
5 //定义允许的最大活动数
6 #define Maxn 100
7
8 //定义活动的类型
9 typedef struct act_Node
10 { int Id; //活动ID
11   int s_Time; //活动开始时间
12   int f_Time; //活动结束时间
13 } ACND;
14
15
16 //对活动按贪心准则排序
17 void Sort(int n,ACND arr[])
18 {
19     ACND *temp;
```

	输入	期望输出	实际输出	
✓	4 1 5 6 2 1 2 3 3 7 4 1 2	2:1-2 1:5-6	2:1-2 1:5-6	✓

通过所有测试 ✓

正确
此次提交得分：20.00/20.00。