

南开大学 2012 年硕士研究生入学考试试题

学 院：034 信息技术科学学院
考试科目：831 程序设计与数据结构
专 业：计算机技术

注意：请将所有答案写在专用答题纸上，答在此试题上无效！

数据结构部分

一、单项选择题（共 20 分）

1. 若 $x=100$, 则与 `while(x)` 语句等价的语句是 ()。(2 分)
 - A. `while(x==0)`
 - B. `while(x==100)`
 - C. `while(x!=100)`
 - D. `while(x!=0)`
2. 在 VC 上执行下列语句后, a 的值是 () (2 分)

```
int a=3;
a+=--a*a;
```

 - A. 12
 - B. 24
 - C. 20
 - D. 6
3. 已知: `int a[3][5]`; 则对数组元素的非法引用是 () (2 分)
 - A. `a[0][2*1]`
 - B. `a[0][0]`
 - C. `a[3][0]`
 - D. `a[0][4]`
4. 假定 w 、 x 、 y 、 z 、 m 均为 `int` 型变量, 如下程序段在 VC 上运行后, m 的值是 ()。(2 分)

```
.....
int m, w=1, x=2, y=6, z=4;
m=(w<x)?w:x;
m=(m<y)?y:x;
m=(m>z)?m:z;
.....
```

 - A. 4
 - B. 3
 - C. 2
 - D. 6
5. 可以用数组名作为字符串变量整体输出的合法的数组说明语句是 ()。(2 分)
 - A. `char a[]="string";`

- B. `int a[5] = {0,1,2,3,4,5};`
C. `char a = "string";`
D. `char a[] = {'1','2','3','4','5'};`
6. 关于输入/输出流, 下列描述正确的是 ()。(2分)
A. `cin` 和 `cout` 是 C++语言中用于输入输出的流类
B. 使用 `cin` 和 `cout` 进行文件读写, 必须首先打开文件
C. `cin` 和 `cout` 可以对任何类型的数据或类对象进行输入输出
D. `cin` 和 `cout` 以运算符重载的方式实现数据的输入输出
7. 下列哪一项是对构造函数和析构函数的正确定义 ()。(2分)
A. `void X::X(); void X::~X();`
B. `X::X(参数); X::~X();`
C. `X::X(参数); X::~X(参数);`
D. `void X::X(参数); void X::~X(参数);`
8. 如果某个类 `TestClass` 的定义中只有下列原型的构造函数: `TestClass:: TestClass(int,int);` 则下列表达式中, 能够正确初始化类 `TestClass` 对象的是 () (2分)
A. `TestClass Obj1; Obj1.TestClass(1, 2);`
B. `TestClass Obj2[10](1, 2);`
C. `TestClass *Obj3 = new TestClass(1, 2);`
D. `TestClass Obj4 = TestClass(1, 2);`
9. 关于函数模板和类模板, 下列描述正确的是 () (2分)
A. 函数模板和类模板的类型参数只能是基本数据类型
B. 类模板的静态成员在定义类模板时创建
C. 可以直接创建类模板的对象并用构造函数进行初始化
D. 函数模板可以重载而且不允许参数类型转换
10. 下列函数中, 不能重载的是 () (2分)
A. 类的成员函数
B. 类的非成员函数
C. 类的构造函数
D. 类的析构函数

二、写出下列程序在 VC 6.0 环境下(X86 平台)的运行结果 (共 25 分)

1. (5分) 从键盘输入 "There are 13 students in SUN Team". 写出程序的运行结果。

```
#include <iostream.h>
#include <string.h>
void main()
{ char c[80];
  int i=0,j=0,k=0;
  cout<<"Please Input Message : ";
  cin.getline(c,80);
  for (int t=0; c[t]!='\0';t++)
```

```

    {
        if (c[t]<='z' && c[t]>='a') i++;
        if (c[t]<='Z' && c[t]>='A') j++;
        if (c[t]<='9' && c[t]>='0') k++;
    }
    cout<<j<<"t"<<j<<"t"<<k<<"t"<<endl;
}

```

2. (5分) 写出程序运行结果。

```

#include <iostream.h>
int a=7,b=5;
void main()
{
    int a=3,b=8;
    while (a<b)
    {
        if ( a=4)  b/=2;
        cout <<"a="<<a<<"b="<<b<<endl;
        if ( a=3)  b+=2;
        a++;
    }
    cout <<"a="<<a<<"b="<<b<<endl;
    cout <<"a="<<::a<<"b="<<::b<<endl;
}

```

3. (5分) 写出程序运行结果。。

```

#include <iostream.h>
void main(){
    int a=1,b=2;
    while (b<8){
        switch (a+b){
            case 3:
                a+=b;
                cout<<"a="<<a<<" b="<<b<<endl;
            case 4: a/=b++;
                b++;
                cout<<"a="<<a<<" b="<<b<<endl;
                break;
            case 5: a++;
                cout<<"a="<<a<<" b="<<b<<endl;
                continue;
        }
    }
}

```

```

        case 6: if (a > 1) a++;
                if (b < 5) b += a++;
                cout << "a=" << a << " b=" << b << endl;
        default: b *= a % 5;
    }
}
cout << "a=" << a << " b=" << b << endl;
}

```

4. (5 分) 写出程序运行结果。。

```

#include <iostream.h>
int *findvalue(int *array, int size, int &index){
    int min = 0;
    for(int i = 1; i < size; i++){
        if (array[i] < array[0]){
            array[0] = array[i];
            min = i;
        }
    }
    index = min;
    return array + min;
}

void main() {
    int b[10] = {99, 5, 12, 54, 77, 66, 13, 23, 9, 76};
    int *minaddr, idx;
    minaddr = findvalue(b, sizeof(b)/sizeof(int), idx);
    cout << "value of index: " << idx << endl;
    cout << "value of *pointer: " << *minaddr << endl;
    cout << "value: " << b[idx] << endl;
}

```

5. (5 分) 写出程序运行结果。。

```

#include <iostream.h>
const int maxsize = 6;
template <class T> class stack{
    T data[maxsize];
    int top;
public:
    stack() {
        top = 0;
    }
}

```

```

int lsempy() { if(top==0) return 1; else return 0;}
void push(T a) {
    if(top==maxsize) {
        cout<<"stack overflow!"<<endl;
        return;
    }
    data[top]=a;
    top++;
}
T pop() {
    if(top==0){
        cout<<"An empty stack!"<<endl;
        return 0;
    }
    top--;
    return data[top];
}
};

void main(){
    stack<int> si1,si2;
    cout<<"Please input "<<maxsize<<" integers: ";
    for(int i=1; i<=maxsize; i++) {
        int j;    cin>>j;
        si1.push(j);
        si2.push(2*j);
    }
    do
        cout<<si1.pop()<<","<<si2.pop()<<",";
    while(!si2.lsempy());
    cout<<endl;
}

```

在 VC6.0 环境下执行输入:
Please input 6 integers: 6 5 4 3 2 1

三、根据题意完成下列程序。(共 30 分)

1. (6 分) 下列是一个可以对任何类型数据进行选择排序的函数模板。

```

template <class T>
void SelectExec(T *p,int n){
    for(int i=0;i<n-1;i++){
        (1)
        for ( (2) )if (p[j]<p[temp])temp=j;
    }
}

```

```

        T t=p[i];
        p[i]=p[temp];
        p[temp]=t;
    }
    Print(p,n);
}
template <class T>
Print(T *p,int n){
    for (int k=0;k<n;k++)cout<<_____(3)____<<" ";
    cout<<endl;
}

(1)
(2)
(3)

```

2. (6 分)

函数 find 的功能是：在有 n 个学生信息的数组 s 中查找名为 stu_name 的学生，若找到，函数返回数组下标，否则，函数返回-1。

```
#include <iostream.h>
```

```
#include <string.h>
```

```
struct data
```

```
{ int id;
```

```
    char name[20];
```

```
    double score;
```

```
} students[100];
```

```
int findbook(data s[],int n,char stu_name[])
```

```
{
```

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

```
        if(_____(1)____)
```

```
    _____(2)_____;
```

```
    _____(3)_____;
```

```
}
```

(1)

(2)

(3)

3. (6 分)

指针与字符串。函数 stat 的功能是统计字符串中数字字符、英文字母以及其它字符的数量，

请根据主函数完善 stat 函数。

```
#include<iostream.h>
```

```
void stat( _____ (1) _____ ){  
    while(*str!='\0'){  
        if(*str >='0' && *str <='9')  
            n++;  
        else {  
            if( _____ (2) _____ )  
                a++;  
            else  
                o++;  
        }  
        _____ (3) _____ ;  
    }  
}
```

```
void main()
```

```
{  
    int num=0,aphlet=0,other=0;  
    stat("abcdef1234^&*(",num,aphlet,other);  
    cout<<"num="<<num<<endl<<"aphlet="<<aphlet<<endl;  
    cout<<"other="<<other<<endl;  
}
```

(1)

(2)

(3)

4. (6分)。

设计复数类，重载运算符“<<”实现对复数的输出。请完善该程序。

```
#include<fstream.h>
```

```
class complex {
```

```
    double r;
```

```
    double i;
```

```
public:
```

```
    complex(double r0=0, double i0=0) {
```

```
        r=r0;i=i0;
```

```
    }
```

```
    _____ (1) _____ ;
```

```
};
```

```
ostream& operator << (ostream& out, complex com) {
```

```
    out<<"("<<com.r<<" "<<com.i<<"")<<endl;
```

```

        _____(2)_____ ;
    }
    void main() {
        complex c1(1,1), c2(2,3);
        _____(3)_____ ; //输出复数 c1 和 c2
    }
    (1)
    (2)
    (3)

```

5. (6 分)。

用递归函数实现, 当输入 n 为偶数时, 输出 $1/2+3/4+\dots+(n-1)/n$, n 为奇数时输出 $1/1+1/3+\dots+1/n$ 。

```

double s(long n)
{
    if (____1____)
    {
        if (n==2) return(1.0/2);
        else return(____2____);
    }
    else
    {
        if (n==1) return (1.0/n);
        else return (____3____);
    }
}
(1)
(2)
(3)

```

数据结构部分

四、单项选择题 (共 24 分)

- 下列排序算法中, _____、_____ 可以作为基数排序每个阶段的排序算法, _____、_____ 不可以。(4 分)
 - 起泡排序
 - Shell 排序
 - 箱子排序
 - 快速排序
- 序列 86, 77, 57, 12, 25, 30 为最大堆, 将 90 插入其中, 得到的结果为_____。(3 分)
 - 86, 77, 57, 12, 25, 30, 90
 - 90, 77, 86, 12, 25, 30, 57

C. 90, 77, 86, 12, 25, 57, 30

D. 90, 86, 77, 12, 25, 57, 30

3. 对序列 86, 57, 77, 12, 25, 30 进行排序（整理为升序），若前三趟排序结果如下

第一趟：57, 77, 12, 25, 30, 86

第二趟：57, 12, 25, 30, 77, 86

第三趟：12, 25, 30, 57, 77, 86

则采用的排序方法可能是_____。（3分）

A. Shell 排序

B. 选择排序

C. 箱子排序

D. 起泡排序

4. 设二叉树 T 的中序遍历结果为 2 1 4 5 3，后序遍历结果为 2 5 4 3 1，则 T 的先序遍历结果为_____。（3分）

A. 1, 2, 3, 4, 5

B. 2, 3, 1, 4, 5

C. 3, 2, 1, 4, 5

D. 有多种可能，不唯一

5. 飞机登机时，会请老幼病残乘客和头等舱/VIP 乘客先登机，如果要设计飞机登机的仿真程序，应采用哪种数据结构？_____。（2分）

A. FIFO 队列

B. 堆

C. 栈

D. 树

6. 高度为 2 的 5 阶 B 树，最多包含_____个关键字，最少包含_____个关键字。（4分）

A. 4

B. 5

C. 6

D. 24

E. 25

F. 26

7. 7 个顶点的最大堆，排在降序第 3 位的元素有几个可能的位置_____。（2分）

A. 3

B. 4

C. 5

D. 6

8. 二叉树 T 中，有两棵非空子树的节点数为 20，有且只有一棵非空子树的节点数为 10，则叶节点数为_____。（3分）

A. 21

B. 30

C. 40

D. 51

五、简答题（共 40 分）

1. （12 分）对序列 503, 17, 512, 908, 170, 897, 275, 653, 612, 154, 765, 94 进行堆排序，给出排序过程和最终结果。
2. （16 分）设一个 Hash 表的地址区间为 0-15，哈希函数为 $H(K)=K \bmod 16$ 。采用线性探测法处理冲突，请将关键字序列 17, 170, 275, 154, 94, 79, 14, 66, 20, 84, 45, 73, 13 依次存储到哈希表中，画出结果，并计算平均查找长度。这样的 Hash 函数设计好不好？为什么？如果不好，你是否可以设计出更好的 Hash 函数？
3. （12 分）将关键字 12, 70, 33, 65, 24, 56, 48, 92, 86, 33 按此顺序插入到空 AVL 搜索树中，画出每一步的结果和最终结果。

六、算法设计题（11 分）

设计函数，求最大二叉堆中的最小元素，设计优化策略，避免不必要的搜索步骤。其中， n 为堆中元素总数， a 为保存堆的数组，返回值为最小元的数组下标，如必要可设计辅助函数。

`int min_in_max_heap(int a[], int n)`

