华侨大学 2012 年硕士研究生入学考试专业课试卷

（答案必须写在答题纸上）

招生专业 计算机技术

科目名称 数据结构与 C++ **科目代码** 850

第一部分 C++程序设计（共 75 分）

一、是非题（共 10 分，每题 1 分）

1. 一个数据对象的内存地址称为该数据对象的指针 。
2. 指针函数就是函数的类型为指针数据类型。
3. 程序设计语言的语义表示语言结构的书写规则。
4. 数组的存储类别不可以是 register。
5. switch 语句的 case constant-exp 只作为语句的入口标号，本身并不能改变控制流程，也即无法控制跳过 switch 语句的其它分支。
6. 因为函数内部声明的数据存放在函数栈区，所以其作用域是局部的。
7. 数组的体积是静态确定的，而向量的体积（容器的大小）是可以动态改变的。
8. 类的析构函数不但可以重载，而且还可以设置默认参数。
9. 成员函数是类的操作的实现，一个类可以包含多个成员函数。这些函数可以由类的所有对象所共享。
10.为了保证类的封装性，C++规定类的所有成员都只能在类的内部定义。

二、填空题（共 10 分，每空 1 分）

1. 设一个 C++程序由 f1、f2、f3、f4 和 main 五个函数构成，f1 中分别调用了 f2 和 f3，f2 中调用了 f1,则对于 f1 的递归调用形式，我们可以说 (1) 。
2. I/O 流是标准字符设备上的一系列字符组成的 (2) ，标准字符设备分为 (3) 和 (4) 分别用 (5) 和 (6) 表示。
3. 外部函数应声明成类的 （7） ，才能访问类的类的私有成员。
4. 使用友元函数可以提高程序执行效率的理由是 （8） 。
5. 表达式的语义涉及两方面：其一， （9） 。 第二，执行过程中所要遵循的 （10） 、求值次序及优先级规则。

三、阅读以下程序并给出执行结果（10 分）

1.（3 分）

```
#include <iostream>
#include <string>
using namespace std;
void main()
{
    char a[100],t;
    int i,j,k;

    cin >>a;
    k=strlen(a);
    for (i=0;i<=k-2;i+=2)
```

```
        for (j=I+2;j<k;j+=2)
            if (a[i]>a[j])
            {
                t=a[i];a[i]=a[j];a[j]=t;
             }
    cout <<a<<endl;
}
```

输入：kajbacl

输出：?

2.（3 分）

```
#include <iostream>
using namespace std;
class Tc
{
    int v1,v2;
  public:
    Tc(int x=0,int y=0):v1(x),v2(y)
    {cout <<"This is a constructor!v1+v2="<<v1+v2<<endl;}
    ~Tc() {cout<<"This is a destructor!"<<endl;}
};
void main()
{
    Tc x,y(10,20),z(y);
}
```

输出：?

3.（4 分）

```
#include <iostream>
using namespace std;
class Bclass
{public:
  Bclass(int i, int j)
  {x=i;y=j;}
  virtual int fun(){return 0;}
protected:
int x,y;
};
class Iclass:public Bclass
{public:
  Iclass(int i,int j,int k):Bclass(i,j)
  {z=k;}
  int fun(){return(x+y+z)/3;}
private:
  int z;
};
```

```
void main()
{ Iclass obj(2,4,10);
   Bclass p1=obj;
   cout<<p1.fun()<<endl;
   Bclass & p2=obj;
   cout<<p2.fun()<<endl;
   cout <<p2.Bclass::fun()<<endl;
   Bclass *p3=&obj;
   cout<<p3->fun()<<endl;
}
```

输出：？

四、阅读以下程序或函数，简要叙述其功能（10 分）

1.（3 分）

```
int f1(int a[],int n,int key)
{
    int low,high,mid;
    low=0;
    hight = n-1;
    while (low<=high)
    {
        mid =(low+high)/2;
        if (key<a[mid])
            high=mid-1;
        else
            if (key >a[mid])
                low=mid+1;
            else
                 return mid;
    }
    return -1;
}
```

功能：？

2.（3 分）

```
#include <iostream>
#include <iomanip>
#include <cmath>
using namespace std;
void main()
{
     int i,m,m1,sqrtm,n1,n2;
     int l=0;
     cin>>n1>>n2;
     for (m=n1;m<=n2;m+=2)  {
```

```
          if (m!=2) m1=m-1;
          else m1=m;
          sqrtm=sqrt(m1);
          for (i=2;i<=sqrtm;i++) {
               if(m1%i==0)
               break;
          }
          if(i>sqrtm) {
              if (l++%10==0)
                  cout <<endl;
              cout <<setw(5) << m1;
          }
      }
}
```

功能：？

3.（4 分）

简述 Matrix 类的功能。

```
#include<iostream>
using namespace std;
class Matrix
{
   public:
     Matrix();
     friend Matrix operator+(Matrix &,Matrix &);
     friend ostream& operator<<(ostream&,Matrix&);
     friend istream& operator>>(istream&,Matrix&);
   private:
     int mat[2][3];
  };
Matrix::Matrix()
{
   for(int i=0;i<2;i++)
      for(int j=0;j<3;j++)
           mat[i][j]=0;
}
Matrix operator+(Matrix &a,Matrix &b)
{
   Matrix c;
   for(int i=0;i<2;i++)
      for(int j=0;j<3;j++)
           c.mat[i][j]=a.mat[i][j]+b.mat[i][j];
      return c;
}
istream& operator>>(istream &in,Matrix &m)
```

```
{
    cout<<"input value of matrix:"<<endl;
        for(int i=0;i<2;i++)
            for(int j=0;j<3;j++)
                in>>m.mat[i][j];
    return in;
}
ostream& operator<<(ostream &out,Matrix &m)
{
  for (int i=0;i<2;i++)
    for(int j=0;j<3;j++)
      out<<m.mat[i][j]<<" ";
  out<<endl;
  return out;
}
```

五、编程题（35 分）

1. 编写计算 1！+3！+5！+.....+n!的程序，要求用递归函数实现阶乘运算。（10 分）

2. 设计一个满足如下要求的日期类。（25 分）
（1）设置日期；
（2）按 day/moth/year 的格式输出日期；
（3）输出对当前日期加上一天后的日期。

第二部分 数据结构（共 75 分）

六、单项选择题（每小题 2 分，共 20 分）

1. 下面算法的时间复杂度为（）。

```
int s=0;
for(int i=0;i<m;i++)
  for(int j=0;j<n;j++)
      s+=a[i][j];
```

A) O(m)　　B) O(n)　　C) O(m*n)　　D) O(m+n)

2. 在一个长度为 n 的顺序表中，删除第 i 个位置（1≤i≤n）上的元素时，需要移动（）个元素。

A) n-i　　B) n-i+1　　C) n-i-1　　D) i

3. 若栈采用顺序存储方式存储，现两栈共享空间 V[1..m]，top[i]代表第 i 个栈(i =1,2)的栈顶，栈 1 的底在 v[1]，栈 2 的底在 V[m]，则栈满的条件是（）。

A) |top[2]-top[1]|==0　　B) top[1]+1==top[2]
C) top[1]+top[2]==m　　D) top[1]==top[2]

4. 3 个结点的二叉树共有（）种形态。

A) 3　　B) 4　　C)5　　D) 6

5. 一个无向连通图的生成树是含有该连通图的全部顶点的（）。

A.极小连通子图　　B.极小子图　　C.极大连通子图　　D.极大子图

6. 堆的形状是一棵（）。

A）二叉排序树　　B）满二叉树　　C）完全二叉树　　D)平衡二叉树

7.在含有 n 个顶点、e 条边的无向图的邻接矩阵中，非零元素的个数为（）。

A）e　　B）2e　　C）n^2-e　　D）n^2-2e

8. 设有 50 个元素，用折半查找法进行查找时，在查找成功的情况下，最大比较次数是（）。

A）4　　B）5　　C)6　　D)7

9. 设有一个用开放定址法的线性探测再散列处理冲突的哈希表如下：

0	1	2	3	4	5	6	7	8	9	10
		13	25	80	16	17	6	14		

哈希函数为 H（key）=key%11，若要查找关键字 14 的记录，所需的比较次数是（）。

A）3　　B）6　　C)7　　D）9

10. 对初始状态为递增序列的表按递增顺序排序，最费时间的是（）算法。

A)堆排序　　B）归并排序　　C)插入排序　　D）快速排序

七、算法分析和阅读题（18 分）

1.（6 分,每空 2 分）设单链表的类型定义如下。list 为带头结点的单链表头指针，下面算法实现求指定序号为 order（order≥1）的结点的位置并返回，请填空加以完成。

```
typedef struct Node{
     char name[10];
     int age,score;
     struct Node* next;
}*LinkList;
LinkList  fun1(LinkList& list,int order) {
  if(! (order>=1 && order<=Get_length(list)) ){
       // 函数 Get_length(list)返回链表的长度
        cout<<"Error order found\n";
        return NULL;
  }
  LinkList t= __(1)__ ;
  int num=1;
  while( (2) ){
      num++;
      (3) ;
  }
  return t ;
```

```
}
```

2.（6 分） 阅读下面算法，指出该算法的功能。

```
Status  fun2(char  *str){
   Stack T;  Queue Q;
   char ch1, ch2;     int i ;
   InitStack（T）;
   InitQueue(Q);
   for  ( i = 0;   str[i] != '\0';   i++) {
       Push( T,  str[i] );
       EnQueue( Q, str[i] );
   }
   for ( i =1;  i <= StackLength(T)/2 ;  i++ ){
      //StackLength(T)返回栈 T 的长度
      Pop(T, ch1) ;  DeQueue( Q, ch2);
      if ( ch1 != ch2 )   return   false;
   }
   return  true;
}
```

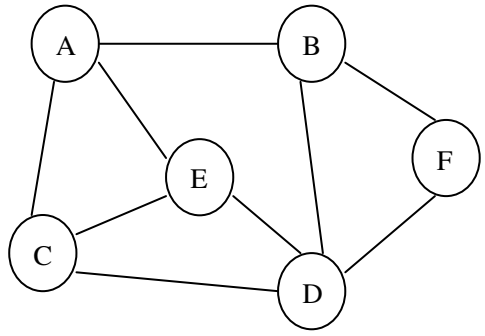
3.（6 分）设二叉树 T 以二叉链表为存储结构，指出下面算法的功能。

```
   int  fun3(BiTree T){
      if (!T )  return 0;
      if (T->lchild == NULL && T->rchild == NULL )  return 1;
      else  return fun3(T->lchild) + fun3(T->rchild)+1;
   }
```

八、简答题（13 分）

1.如何理解链表中的头指针和头结点？(4 分)

2.（6 分）若有如下无向图 G：
（1）试画出 G 的邻接表表示，要求邻接表中的表结点按顶点序号从小到大排列；（2 分）
（2）根据你所给出的邻接表，分别给出从顶点 A 出发的深度优先搜索序列和深度优先搜索生成树。（4 分）



无向图 G

3.（3 分） 设有一组关键字{25,57,48,37,12,92,6,86,15,33,18,20}，请给出快速排序的第一趟结果。

九、算法设计题（24 分）

1．分别编写非递归与递归算法，输出不带头结点的单向链表 L 的各个元素值（11 分）

2．设二叉排序树 bt 以二叉链表为存储结构，试设计算法删除二叉排序树 bt 中值最大的结点。（13 分）