数据结构部分

选择题

1、下面程序的空间复杂度为

```
long long Factorial(size_t N)
{
    return N < 2 ? N : Factorial(N - 1) * N;
}
```

A、$O(n)$

B、$O(log_2 n)$

C、$O(n^2)$

D、$O(1)$

2、下面程序的时间复杂度为

```
int BinarySearch(int* a, int n, int x)
{
    assert(a);

    int begin = 0;
    int end = n-1;
    while (begin < end)
    {
        int mid = begin + ((end-begin)>>1);
        if (a[mid] < x)
            begin = mid+1;
        else if (a[mid] > x)
            end = mid;
        else
            return mid;
    }

    return -1;
}
```

A、$O(n)$

B、$O(log_2 n)$

C、$O(n^2)$

D、$O(nlog_2 n)$

3、在一个具有 n 个结点的有序单链表中插入一个新结点并仍然保持有序的时间复杂度是

A、$O(n)$

B、$O(log_2 n)$

C、$O(n^2)$

D、$O(nlog_2 n)$

4、设一个链表最常用的操作是在末尾插入结点和删除尾结点，则选用（ ）最节省时间。

A、单链表
B、单循环链表
C、带尾指针的单循环链表
D、带头结点的双循环链表

5、一个单向链表队列中有一个指针p，现要将指针r插入到p之后，该进行的操作是

A、`p->next=p->next->next`
B、`r->next=p;p->next=r->next`
C、`r->next=p->next;p->next=r`
D、`r=p->next;p->next=r->next`

6、若进栈序列为 1,2,3,4 ，进栈过程中可以出栈，则下列不可能的一个出栈序列是

A、1,4,3,2
B、2,3,4,1
C、3,1,4,2
D、3,4,2,1

7、一组记录排序码为(5 11 7 2 3 17)，现将其整理成大堆后的序列为

A、(17 11 7 2 3 5)
B、(17 11 7 5 3 2)
C、(17 7 11 3 5 2)
D、(17 7 11 3 2 5)

8、一棵完全二叉树的节点数位为531个，那么这棵树的高度为

A、11
B、10
C、8
D、12

9、关于排序，下面说法不正确的是

A、快排时间复杂度为$O(nlog_2 n)$，空间复杂度为$O(log_2 n)$
B、归并排序是一种稳定的排序，堆排序和快排均不稳定
C、序列基本有序时，快排退化成冒泡排序，直接插入排序最快
D、归并排序空间复杂度为$O(n)$, 堆排序空间复杂度的为$O(log_2 n)$

10、设一组初始记录关键字序列为(65, 56, 72, 99, 86, 25, 34, 66)，则以第一个关键字65为基准而得到的一趟快速排序结果是

A、34，56，25，65，86，99，72，66
B、25，34，56，65，99，86，72，66
C、34，56，25，65，66，99，86，72
D、34，56，25，65，99，86，72，66

问答题

1、简述如何将两个有序链表合并成一个新的有序链表

2、以大堆为例，简述向下调整算法的执行方式

3、已知二叉树的前序遍历结果是ABDGHJCEIKF，中序遍历是GDJHBAKIECF，请还原这棵树并写出其后序遍历结果。

4、写出任意一种排序算法

C++部分

选择题

1、`print()`函数是一个类的常成员函数，它无返回值，下列表示中正确的是

A、`const void print();`
B、`void const print();`
C、`void print() const;`
D、`void print(const);`

2、以下关于纯虚函数的说法,正确的是

A、声明纯虚函数的类不能实例化
B、声明纯虚函数的类成虚基类
C、子类必须实现基类的
D、纯虚函数必须是空函数

3、下列有关this指针使用方法的叙述正确的是

A、保证基类保护成员在子类中可以被访问
B、保证基类私有成员在子类中可以被访问
C、保证基类共有成员在子类中可以被访问
D、保证每个对象拥有自己的数据成员，但共享处理这些数据的代码

4、下列情况中，不会调用拷贝构造函数的是

A、用一个对象去初始化同一个类的另一个新对象时
B、将类的一个对象赋值给该类的另一个对象时
C、函数的形参对象，调用函数进行形参和实参结合时
D、函数的返回值是类的对象，函数执行返回调用时

5、请将下列构造函数补充完整，使得程序的运行结果是5

```
#include<iostream>
using namespace std;

class Sample{
public:
    Sample(int x)
    {
        ________
    }
    ~Sample()
```

```
    {
        if(p) delete p;
    }
    int show()
    {
        return *p;
    }
private:
    int*p;
};

int main()
{
    Sample S(5);
    cout<<S.show()<<endl;
    return 0;
}
```

A、`*p=x;`
B、`p=new int(x);`
C、`*p=new int(x);`
D、`p=&x;`

6、关于内联函数（inline）说法错误的是

A、不是任何一个函数都可定义成内联函数
B、内联函数的函数体内不能含有复杂的结构控制语句
C、递归函数可以被用来作为内联函数
D、内联函数一般适合于只有1~5行语句的小函数

7、关于"深拷贝"，下列说法正确的是

A、深拷贝是针对成员中进行了动态内存分配的指针的
B、深拷贝的拷贝方式是直接拷贝指针的值
C、深拷贝只在拷贝构造的时候执行
D、以上说法都是错误的

8、若要对data类中重载的加法运算符成员函数进行声明，下列选项中正确的是

A、`Data operator+(Data);`
B、`Data operator(Data);`
C、`operator+(Data,Data);`
D、`Data+(Data);`

9、下面关于一个类的静态成员描述中,不正确的是

A、静态成员变量可被该类的所有方法访问
B、该类的静态方法只能访问该类的静态成员函数
C、该类的静态数据成员变量的值不可修改
D、子类可以访问父类的静态成员

10、当一个类对象的生命周期结束后，关于调用析构函数的描述正确的是

A、如果派生类没有定义析构函数，则只调用基类的析构函数
B、如果基类没有定义析构函数，则只调用派生类的析构函数
C、先调用派生类的析构函数，后调用基类的析构函数
D、先调用基类的析构函数，后调用派生类的析构函数

程序阅读题

1、阅读下列程序，写出执行结果：

```
#include<iostream>
using namespace std;
char fun(char x, char y)
{
    if (x < y)
        return x;
    return y;
}

int main()
{
    int a = '1', b = '1', c = '2';
    cout << fun(fun(a, b), fun(b, c));
    return 0;
}
```

2、阅读下列程序，写出执行结果：

```
#include <iostream>
using namespace std;
int f(int n)
{
    if (n==1)
        return 1;
    else
        return (f(n-1)+n*n*n);
}

int main()
{
    int s=f(3);
    cout<<s<<endl;
    return 0;
}
```

3、阅读下列程序，写出执行结果：

```
#include<iostream>
using namespace std;

class B0//基类B0声明
{
public://外部接口
    virtual void display()//虚成员函数
    {
        cout<<"B0::display0"<<endl;
    }
};

class B1:public B0//公有派生
{
public:
```

```
    void display() { cout<<"B1::display0"<<endl; }
};

class D1: public B1//公有派生
{
public:
    void display(){ cout<<"D1::display0"<<endl; }
};

void fun(B0 ptr)//普通函数
{
    ptr.display();
}

int main()//主函数
{
    B0 b0;//声明基类对象和指针
    B1 b1;//声明派生类对象
    D1 d1;//声明派生类对象
    fun(b0);//调用基类B0函数成员
    fun(b1);//调用派生类B1函数成员
    fun(d1);//调用派生类D1函数成员
}
```

编程题

完成一个分数类，支持long long范围内分数的加法和乘法运算，至少包含这些内容：

```
class fraction {
    long long m_numerator;  //分子
    long long m_denominator;//分母

public:
    fraction(long long numerator, long long denominator);
    fraction operator+(fraction const& o) const;
    fraction operator*(fraction const& o) const;

    friend std::ostream& operator<<(std::ostream& os, fraction const& o);
};
```

※注意输出时必须是带分数，且为最简形式。提交时进需要提交类即可，以fraction.cpp的形式发到群里。