

东南大学考试卷（A 卷）

课程名称 C++程序设计(下)

考试学期 09-10-3

得分

适用专业

考试形式 闭卷

考试时间长度 120 分钟

一、选择题（每题 1 分，共 10 分）

1. 在数组 `int array[4][4]={ {45,1,8},{9,28},{11,39,12,76},{64}}` 中，`array[3][1]` 的值是
A. 0 B. 39 C. 8 D. 11
2. 设有变量定义：`int num=20; int *ip; double *dp;` 下面赋值不正确的是____。
A. `ip=#` B. `*ip= #` C. `dp=ip;` D. `*dp=*ip;`
3. 设有变量定义如下：`int k=10;`
`int a=20, m=&k, *p1=&k, *p2=&m;`
执行 `a= p1==&m;` 后 `a` 的值是____。
A. 10 B. 20 C. 0 D. 1
4. 关于通过 `new` 运算符动态创建的对象数组，下列判断中____是错误的。
A. 动态创建的对象数组只能调用默认构造函数
B. 动态创建的对象数组必须调用 `delete []` 动态撤销
C. 动态创建的对象数组的大小必须是常数或常变量
D. 动态创建的对象数组没有数组名
5. 顺序表不具有的特点是____。
A. 元素的存储地址连续 B. 存储空间根据需要动态开辟，不会溢出
C. 可以直接随机访问元素 D. 插入和删除元素的时间开销与位置有关
6. 假设一个对象 `Ob1` 的数据成员是指向动态对象的指针，如果采用浅复制的方式复制该对象得到对象 `Ob2`，那么在析构对象 `Ob1` 和对象 `Ob2` 时会____的问题。
A. 有重复释放 B. 没有 C. 内存泄漏 D. 动态分配失败
7. 假设对 5 个元素 A、B、C、D、E 进行压栈或出栈的操作，压栈的先后顺序是 ABCDE，则出栈的先后顺序不可能是____。
A. ABCDE B. EDCBA C. EDBCA D. BCAED
8. 下列关于虚函数的判断中，____是错误的。
A. 类成员函数前用 `virtual` 修饰的称为虚函数。
B. 只有通过指向派生类对象的基类类型的指针变量或引用来调用基类的虚函数，才能实现动态的多态性。
C. 构造函数一般不定义为虚函数。
D. 派生类中定义的函数名与基类的虚函数名相同就可以实现覆盖。
9. `cin`、`cout`、`cerr` 和 `clog` 四个流对象中有____个是全局标准流对象。
A. 1 B. 2 C. 3 D. 4
10. 打开一个二进制文件时必须采用____模式。

A. ios::in B. ios::out C. ios::binary D. ios::app

二、填空题 (每空 1 分, 共 20 分)

1. 面向对象程序设计具有的特点: _____、_____和_____。
2. C++中的多维数组用的是一个___的定义, 即多维数组的基本定义是以___ 作为元素构成的数组。
3. 通过 new 运算符动态创建的对象存放在_____中。
4. 默认复制构造函数是按成员复制, 称为_____。
5. 为实现代码的____, 通常采用模板, 它把数据类型改为一个____, 称为____程序设计。
6. 类的继承方式有: _____、_____和_____。如果不显示给出访问控制关键字, 则默认继承方式为_____。
7. 以下程序中 Sort() 函数是顺序表的成员函数。

```
template <typename T,int size> void Orderedlist<T,size>::Sort(){
    int i,j,k;
    T temp;
    for(i=0;i<last;i++){
        k=i;    temp=slist[i];
        for(j=i;j<=last;j++)
            if(slist[j]<temp){
                k=j;
                temp=slist[j];
            }
        if(k!=i){
            temp=slist[i];
            slist[i]=slist[k];
            slist[k]=temp;
        }
    }
}
```

该成员函数用到的是_____排序。

若未排序前的数据为: 55 34 98 21 76 43

外层 for 循环执行了第一次后的数据顺序为: _____

8. void a(node *p,Datatype x){

node *q=new node;

q->info=x;

q->link=p->link;

```
p->link=q;
```

```
}
```

以上 a 函数实现的功能是 _____。

9. 通过表达式_____可以判断文件流 fd 的读写指针是否到达文件的尾部。

10. 在_____文件中存取的最小信息单位为字符，而_____文件中存取的最小信息单位为字节。

三、阅读程序, 根据要求填空: (共 40 分)

1. 写出下面程序的输出结果 (每空 2 分, 共 4 分)

```
#include <iostream>
using namespace std;
int max_value(int *arr, int n)
{
    int i, max=arr[0];
    for(i=0; i<n; i++)
        if (arr[i]>max)
            max=arr[i];
    return (max);
}
int main()
{
    int a[3][4] = {{1,3,6,7},{2,4,6,8},{15,17,34,12}};
    int *b=(int*)a;
    cout<<b[7]<<endl;
    cout<<max_value(b,3*4)<<'\n';
    return 0;
}
```

输出结果: _____

2. 指出以下程序的错误: (每空 2 分, 本题 10 分)

```
#include <iostream>
using namespace std;
class Printer
{
    int PrintValue;
    int PrintPages();
public:
```

```

void Printer(int value);
int PrintDocuments();
~Printer();
~Printer(int value);
};
.....//成员函数在类外的定义在此处未具体写出。(略)
int main()
{
    Printer HP, Canon(4);
    Canon.PrintPages();
    return 0;
}

```

错误之处:

3. 写出下面程序的输出结果（每空 2 分，本题 4 分）

```

#include<iostream>
using namespace std;
void swap1(double d1,double d2){
    double temp;
    temp=d1;d1=d2;d2=temp;
}
void swap2(double *d1,double *d2){
    double temp;
    temp=*d1;*d1=*d2;*d2=temp;
}
int main(void){
    double x=1.0, y=2.0 ;
    swap1(x,y);
    cout<<"x="<<x<<"\t"<<"y="<<y<<endl;
    swap2(&x,&y);
    cout<<"x="<<x<<"\t"<<"y="<<y<<endl;
}

```

```
    return 0;
}
```

输出结果: _____

4. 写出下面程序的输出结果（每空 1~3 分，本题 9 分）

```
#include <iostream>
using namespace std;
class stack{
    int *rep;
    int size,top;
public:
    stack(int n=10):size(n)//构造函数
    {
        cout<<"Initial Constructor"<<endl;
        rep=new int[size];
        top=-1;
    }
    stack(stack &s):size(s.size)//复制构造函数，需深复制。
    {
        cout<<"Copy Constructor"<<endl;
        rep=new int[size];
        for (int i=0;i<size;i++)
            rep[i]= s.rep[i];
        top=s.top;
    }
    ~stack()
    {
        cout<<"Destructor"<<endl;
        delete [] rep;
    }
    void push(int a) {rep[++top]=a;    }
    int  pop() { return rep[top--]; }
    bool isEmpty() {return top == -1;}
};

int main()
```

```

{
    stack *ptr=new stack[2];
    for(int i=1;i<5;i++)
    {
        ptr[0].push(i);
        ptr[1].push(i+6);
    }
    stack s2(ptr[0]);
    for(i=0;i<2;i++)
        cout<<s2.pop()<<' ';
    s2.push(ptr[1].pop());
    ptr[0].push(ptr[1].pop());
    s2.push(ptr[1].pop());
    while(!s2.isEmpty())
        cout<<s2.pop()<<' ';
    cout<<endl;
    delete[] ptr;
    return 0;
}

```

输出结果： _____ (1 分)

_____ (1 分)

_____ (1 分)

_____ (3 分)

_____ (1 分)

_____ (1 分)

_____ (1 分)

5. 写出下面程序的输出结果（每空 2 分，本题 8 分）

```

#include<iostream>
#include<string>
using namespace std;
class person
{
    char name[10];
    int age;
public:
    void init1(char *str, int k)

```

```

    {
        strcpy(name,str);
        age=k;
    }
    virtual void disp()
    {
        cout<<"name: "<<name<<endl;
        cout<<"age: "<<age<<endl;
    }
};

class student: public person
{
    int num;
    double avg;
public:
    void init2(int i,double f) {num=i; avg=f;}
    void disp()
    {
        cout<<"num: "<<num<<endl;
        cout<<"avg: "<<avg<<endl;
    }
};

int main()
{
    person A,*p;
    student B;
    A.init1("王明",19);
    p=&A;
    p->disp();
    B.init2(20090332,97.5);
    p=&B;
    p->disp();
    return 0;
}

```

输出结果: _____

6. 写出下面程序的输出结果（每空 1~2 分，共 5 分）

```
#include <iostream>
using namespace std;
template <typename T> T add(T a,T b)
{
    return a+b;
}
void main()
{
    cout<<add<char>('a',2)<<endl;
    cout<<add<int>(3.5,2.9)<<endl;
    cout<<add(1.1,2.1)<<endl;
}
```

输出结果： _____（2 分）

_____（2 分）

_____（1 分）

四、完善程序（13、14 空每空 1 分，其余每空 2 分，共 30 分）

1. 建立线性表模板，能够按照用户要求的元素个数（缺省为 10 个元素）创建数组、实现元素的有序插入（即将元素插入在有序数组适当位置，保持数组有序）、降序排列数组的二分查找以及数组打印。主函数中将模板实例化，创建一个整型数组，通过调用有序插入函数建立一个降序排列的数组，并测试二分查找功能。程序如下：

```
#include<iostream>
using namespace std;
template<typename T>
class seqlist{
    T *slist;
    int last;
    int maxsize;
public:
    seqlist(int n=10){
        maxsize=n;
        slist=new T[n];
    }
};
```



```

        last=-1;
    }
    ~seqlist(){    if(slist)_____ (1) _____ }
void print(){
    for(int i=0;i<=last;i++) cout<<slist[i]<<" ";
    cout<<endl;
}
void insertOrder(T);
int bisearch(T key);
};
template<typename T>
_____ (2) _____{
    for(int i=last; i>=0 && slist[i]<key ;i--) slist[i+1]=slist[i];
    _____ (3) _____=key;
    last++;
}
template<typename T>
int seqlist<T>::bisearch(T key){
    int low=0,high=last,mid;
    while(low<=high){
        mid=(low+high)/2;
        if(key<slist[mid])low=mid+1;
        else if(key>slist[mid]) high=mid-1;
        else break;
    }
    if(low>high)_____ (4) _____ ;
    return mid;
}
int main(){
    seqlist<int> a;
    int k,p;
    cout<<"input 8 integers:"<<endl;
    for(int i=0;i<8;i++){
        cin>>k;
        _____ (5) _____;
        //建立降序排列数组
    }
}

```

```

    }
    a.print();
    cout<<"input an integer to be searched:"<<endl;
    cin>>k;
    p=a.bisearch(k);
    if(p==-1) cout<<k<<" is not in the array."<<endl;
    else cout<<"a["<<p<<"]="<<k<<endl;
    return 0;
}

```

2、建立单链表模板类，并实例化为字符串链表，测试建立链表、查找结点并删除结点等功能。

```

#include<iostream>
#include<string>
using namespace std;
template<typename T>class List;
template<typename T>class Node{
    T data;
    Node* next;
public:
    Node(){next=NULL;}
    Node(T d){data=d; next=NULL;}
    friend class List<T>;
};
template<typename T>
class List{
    Node<T> *head,*tail;
public:
    List(){head=tail=new Node<T>();}    //构造空链表，生成一个空结点
    ~List(){makeEmpty(); delete head;}    //析构函数
    void makeEmpty();    //清空链表，只留下一个表头的空结点
    Node<T>* find(T key);    //搜索值为 key 的结点，返回该结点地址
    void printList();    //打印链表
    Node<T>* createNode(T k){Node<T>* t=new Node<T>(k); return t;}
    //建立数据结点
    void insertRear(Node<T> *p){_____ (6) _____; tail=p;}

```

```

//将 p 结点插入在当前链表尾
void deleteNode(Node<T>* p); //删除 p 结点
};

template<typename T>
void List<T>::printList(){
    Node<T> *t=head->next;
    while(t){
        cout<<t->data<<" ";
        t=t->next;    }
    cout<<endl;
}

template<typename T>
void List<T>::makeEmpty(){
    Node<T> *t;
    while(head->next){
        t=head->next;
        _____ (7) _____;
        delete t;
    }
    tail=head;
}

template<typename T>
Node<T>* List<T>::find(T key){
    Node<T> *t=head->next;
    while(_____ (8) _____) t=t->next;
    return t;
}

template<typename T>
void List<T>::deleteNode(Node<T>* p){
    Node<T> *t=head->next, *q=_____ (9) _____;
    while(q&&q!=p){t=t->next; q=t->next;}
    t->next=q->next;
    delete q;
}

int main(){

```

```

List<string> list1;
Node<string> *pNode;
string s;
cout<<"input 6 strings:"<<endl;
for(int i=0;i<6;i++){
    cin>>s;
    pNode= list1.createNode(s);        //建立数据结点
    list1.insertRear(pNode);
}
list1.printList();
cout<<"input a string:"<<endl;
cin>>s;
pNode=list1.find(s);
if(pNode)_____ (10) _____;    //删除 pNode 结点
else cout<<"no such string."<<endl;
list1.printList();
return 0;
}

```

3、在抽象类 object 基础上派生圆类和矩形类，它们各自重构抽象类中的 calculate 函数，以实现面积的计算；同时重载“<<”运算符实现对象数据的输出。主函数用基类指针指向不同派生类对象，以实现与所指对象相一致的操作。

```

#include<iostream>
#include<cmath>
using namespace std;
class object{        //抽象类
public:
    _____ (11) _____; //定义 calculate 函数
};
class circle:public object{
    double x,y,r;
    double area;
public:
    circle(double x0=0, double y0=0, double r0=1){x=x0;y=y0;r=r0;}
    _____ (12) _____ {area=3.14*r*r; return area;}
    _____ (13) _____ {

```

```

        os<<('<<c.x<<','<<c.y<<')<<"  r="<<c.r<<"  area="<<c.area<<endl;
        return os;
    }
};

class rectangle:public object{
    double left,bottom,right,top;
    double area;
public:
    rectangle(double a=0, double b=0, double c=0,double d=0){
        left=a;bottom=b;right=c;top=d;
    }
    double calculate(){area=fabs((right-left)*(top-bottom));return area;}
    _____ (14) _____ {
        os<<"left="<<r.left<<','<<"bottom="<<r.bottom<<','<<"right="<<r.right<<','<<"top="<<r.top
<<"  area="<<r.area<<endl;
        return os;
    }
};

int main(){
    _____ (15) _____;
    circle c(1,2,4);
    pObj=&c;
    double area=pObj->calculate();    //计算圆面积
    cout<<c;                          //输出圆参数
    rectangle r(1,1,4,5);
    pObj=&r;
    _____ (16) _____;    //计算矩形面积
    cout<<r;                      //输出矩形参数
    return 0;
}

```

2009~2010-3 东南大学《C++程序设计 II》(A 卷)

参考答案

一、选择题 (每题 1 分, 共 10 分)

1	2	3	4	5	6	7	8	9	10
A	B	D	C	B	A	C	D	D	C

二、填空题 (每空 1 分, 共 20 分)

- 封装性、继承/派生性、多态性
- 嵌套、数组
- 堆/自由存储区/Heap
- 浅复制
- 重用、设计参数、参数化
- 公有继承、私有继承、保护继承 私有继承
- 直接选择 21 34 98 55 76 43
- 在单链表的 p 结点后插入一个数据域为 x 的新结点
- fd.eof()
- 文本 二进制

三、阅读程序, 根据要求填空: (共 40 分)

- 写出下面程序的输出结果 (每空 2 分, 共 4 分)

输出结果: 8

 34

- 指出以下程序的错误: (每空 2 分, 本题 10 分)

错误之处:

类 Printer 的构造函数不能有返回值, 应将 void 删掉;

类 Printer 的析构函数不应有参数;

类 Printer 的析构函数不能够重载;

类对象 HP 在定义时必须带参数;

成员函数 PrintPages() 是私有成员函数, 不能被类的对象访问。

- 写出下面程序的输出结果 (每空 2 分, 本题 4 分)

输出结果: x=1 y=2

 x=2 y=1

- 写出下面程序的输出结果 (每空 1~3 分, 本题 9 分)

输出结果: Initial Constructor (1 分)

 Initial Constructor (1 分)

 Copy Constructor (1 分)

 4, 3, 8, 10, 2, 1, (3 分)

 Destructor (1 分)

 Destructor (1 分)

 Destructor (1 分)

5. 写出下面程序的输出结果（每空 2 分，本题 8 分）

输出结果： name: 王明
age: 19
num: 20090332
avg: 97.5

6. 写出下面程序的输出结果（每空 1~2 分，共 6 分）

输出结果： c (2 分)
5 (2 分)
3.2 (1 分)

四、完善程序（13、14 空每空 1 分，其余每空 2 分，共 30 分）

1. (1) delete []slist;
(2) void seqlist<T>::insertOrder(T key)
(3) slist[i+1]
(4) mid= -1
(5) a.insertOrder(k)
2. (6) tail->next=p
(7) head->next=t->next
(8) t&& t->data!=key 或 t!=NULL&&t->data!=key 或 t!=0&&t->data!=key
(9) t->next
(10) list1.deleteNode(pNode)
3. (11) virtual double calculate()=0
(12) double calculate()
(13) friend ostream &operator<<(ostream &os, circle &c) (1 分)
(14) friend ostream &operator<<(ostream &os, rectangle &r) (1 分)
(15) object *pObj
(16) pObj->calculate()