

2017-2018 学年第 1 学期考试试题

参考答案和评分标准

课程名称 《面向对象程序设计》 考试方式 闭卷

适用专业 2016 级计算机类各专业、英软 考试时间 (120) 分钟

一、判断题 (共 10 分, 每小题 1 分)

题号	1	2	3	4	5	6	7	8	9	10
答案	×	√	×	√	×	×	√	×	√	√

评分标准: 每答对 1 题计 1 分, 答错不计分。

二、单选题 (共 20 分, 每小题 2 分)

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

评分标准: 每答对 1 题计 2 分, 答错不计分。

三、程序分析 (共 30 分, 每小题 5 分)

1、 200,100

2、 68

9
5.5
8.2

3、 12/31/2017
1/1/2018

4、 Part:0
Part:2
Part:3
Whole:5

5、 ctor:4,5
dctor:4,5

6、 <1>A::A()
<1>A::A()
<3>B::B()
<2>A::g()
<4>B::g()

评分标准: 每答对 1 题计 5 分, 否则酌情计分。

四、程序设计 (共 40 分, 每小题 10 分)

1、参考实现如下:

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

```
template <class T>
int seqSearch(T array[], int n ,T &key) {
    for(int i = 0; i < n; i++)
        if (array[i] == key)
            return i;
    return -1;
}
```

```
int main()
{
    const int n=10;
    double a[n],k;
    for(int i=0;i!=n;++i)
        cin>>a[i];
    cout<<"请输入要查找的数:"<<endl;
    cin>>k;
    int t= seqSearch(a,n,k);
    if(t>0)
        cout<<k<<"在数组中的下标是"<<t<<endl;
    else
```

```
cout<<"在数组中没有找到"<<k<<endl;
```

```
}
```

评分标准：语法格式正确计2分，正确实现模板功能计6分，主函数中调用合理计2分，否则酌情计分。

2、参考实现如下：

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

```
class Ball{
private:
    double r;
public:
    Ball(double n=0){ r=n; }
    double Area(){ return 4*3.14*r*r; }
    double Volume(){ return 4*3.14*r*r*r/3; }
    void SetR(double x){ r = x; }
};
```

```
int main()
{
    double n;
    Ball ball;
    cout<<"请输入球的半径："<<endl;
    cin>>n;
    ball.SetR(n);
    cout<<"该球的表面积为："<<ball.Area()<<endl;
    cout<<"该球的体积为："<<ball.Volume()<<endl;
}
```

评分标准：正确实现各成员函数 8 分，主函数实现 2 分，否则酌情计分。

3、参考程序如下（测试主函数可以不写）

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

```
class CInt{
```

```
private:
```

```
    int n;
```

```
public:
```

```
    CInt(int m=0){n=m; }
```

```
    CInt& operator=(const CInt &rhs);    //重载赋值运算符
```

```
    CInt operator+(const CInt &rhs);    //重载加法运算符
```

```
    friend ostream& operator<<(ostream& os,const CInt &rhs);
    //重载流插入运算符 "<<"
```

```
};
```

```
CInt& CInt::operator=(const CInt &rhs)    //重载赋值运算符
```

```
{
    if(this!=&rhs)n=rhs.n;
    return *this;
}
```

```
CInt CInt::operator+(const CInt &rhs)    //重载加法运算符
```

```
{
    CInt c;
    c.n=n+rhs.n;
    return c;
}
```

```
ostream& operator<<(ostream& os,const CInt &rhs)
```

```
//重载流插入运算符 "<<"
```

```
{
    os<<rhs.n;
    return os;
}
```

```
int main()
```

```
{
    CInt a(1),b(2),c;
    c=a+b;
    cout<<a<<"+"<<b<<"="<<c<<endl;
```

```
}
```

评分标准：语法格式正确2分，正确实现赋值运算符函数重载计3分，加法运算符函数重载计2分，流插入运算符函数重载计3分，否则酌情计分。

4、参考程序如下：

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

class Person{
protected:
    string name, sex;
    int age;
public:
    Person(char* pName="noname", char* pSex="male", int nAge=0)
    {   name=pName;
        sex=pSex;
        age=nAge;
    }
    virtual void Print()const
    {   cout<<"name<<"\t"<<"sex<<"\t"<<"age";
    }
    virtual void DoSomething()const=0;
};

class Teacher:public Person{
private:
    string id, department;
public:
    Teacher(char* pName="noname", char* pSex="male", int
nAge=0, char* pId="000", char* pDepartment="no")
:Person(pName, pSex, nAge)
    {   id=pId;
        department=pDepartment;
    }
    void Print()const
    {   Person::Print();
        cout<<"\t"<<"id<<"\t"<<"department<<endl;
    }
}
```

```
void DoSomething()const
{   cout<<"This is a teacher!"<<endl;
}

};

class Student:public Person{
private:
    string num, className;
public:
    Student(char* pName="noname", char* pSex="male", int nAge=0,
char* pNum="000", char* pClassName="201701"):Person(pName,
pSex, nAge)
    {   num=pNum;
        className=pClassName;
    }
    void Print()const
    {   Person::Print();
        cout<<"\t"<<"num<<"\t"<<"className<<endl;
    }
    void DoSomething()const
    {   cout<<"This is a student!"<<endl;
    }
};

int main()
{   Person *p[2];
    p[0]=new Teacher("Lisa", "female", 35, "001", "计算机学院");
    p[1]=new Student("Tom", "male", 19, "170501", "2017 计算机 1");
    for(int i=0; i!=2; ++i)
    {   p[i]->DoSomething();
        p[i]->Print();
        delete p[i];
    }
}
```

评分标准：主函数实现计 2 分，基类实现计 2 分，两个派生类实现各 3 分，否则酌情计分。