

关于关键字 `class` 和 `typename`, 下列表述正确的是 ()。

- 程序中所有的 `class` 都可以替换为 `typename`
 - 程序中所有的 `typename` 都可以替换为 `class`
 - A 和 B 都可以
 - A 和 B 都不可以
2. 关于在调用模板函数时模板实参的使用, 下列表述正确的是 ()。
- 对于模板实参, 如果能从模板函数的实参中获得相同的信息, 则都可以省略
 - 对于模板实参, 如果是参数表中最后的若干个参数, 则都可以省略
 - 对于模板实参, 若能够省略则必须省略
 - 对于常规参数所对应的模板实参, 任何情况下都不能省略
3. 关于类模板, 下列表述中错误的是 ()。
- 用类模板定义一个对象时, 有可能省略实参表
 - 类模板只能有虚拟类型参数
 - 类模板本身在编译中不会生成任何代码
 - 类模板的成员函数都是模板函数
4. 有如下函数模板声明:

```
template<typename T>
T func(T x, T y){ return x*x+y*y; }
```

在下列对 `func` 的调用中错误的是 ()。

- `func(3,5)`
 - `func<>(3,5)`
 - `func(3,5.5)`
 - `func<int>(3,5.5)`
5. 下列说法不正确的是 ()。
- 如果在程序中没有调用一个已有模板函数, 则不会生成该函数模板的任何实例
 - 如果在程序中没有定义一个已有模板类的对象, 则不会生成该类模板的任何实例
 - 函数模板的一个实例就是一个函数定义
 - 类模板的一个实例就是一个对象
6. 下列类模板的定义中语法格式不正确的是 ()。
- `template <class T> class Buffer { /* ... */};`
 - `template <typename T> class Buffer { /* ... */};`
 - `template <int N> class Buffer { /*... */};`
 - `template <T> class Buffer { /* ... */};`

7. 有如下程序:

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

template<typename T>
T max(T d1, T d2){ return d1>d2 ? d1 : d2; }

int main(){
    cout<<max(3,5)<<endl;
    cout<<max(5,3)<<endl;
    cout<<max(5.0, max(3.5,5.5))<<endl;
}
```

```
return 0;
```

编译这个程序的过程中，函数模板 max 被实例化的次数是 ()。

- A. 1次 B. 2次 C. 3次 D. 4次
8. 关于类模板的实例化，下列表述错误的是 ()。
- A. 在利用类模板定义模板类对象时，有可能不生成任何实例
- B. 只有为类模板的参数指定了默认值的情况下，定义对象才可以省略模板实参
- C. 在定义模板类对象数组时，将生成与数组元素个数相等的多个模板类实例
- D. 类模板的实例化就是生成一个类定义的过程
9. 已知下面是一个正确的语句：

```
cout<<myFunc<double,9>(30);
```

这表明模板函数 myFunc ()。

- A. 有两个模板形参，有一个函数形参
- B. 至少有两个模板形参，至少有一个函数形参
- C. 有两个模板形参，至少有一个函数形参
- D. 至少有两个模板形参，有一个函数形参
10. 一个模板函数的原型是

```
template<typename T,int num> T f9(T x[]);
```

且变量 y 和 z 定义为

```
double y[250],z=250;
```

则下列对该模板函数的调用中正确的是 ()。

- A. f9<250>(y) B. f9<double,z>(y) C. f9<z>(y) D. f9<double,250>(y)
11. 2005 年上半年全国计算机等级考试二级笔试试卷 C++ 语言程序设计。
- 关于模板，下列表述中错误的是 ()。

- A. 模板声明中的第 1 个符号总是关键字 template
- B. 在模板声明中用 < 和 > 括起来的部分是模板的形参表
- C. 在任何条件下类模板的实参都不可以省略
- D. 在一定条件下函数模板的实参可以省略
12. 已知一个函数模板定义为

```
template<typename T1, typename T2>
T1 FUN(T2 n){ return n*5.0;}
```

若要求以 int 型数据 7 为函数实参调用该模板函数，并返回一个 double 型数据，则正确的调用格式是 ()。

- A. FUN<double>(7) 或 FUN<double, int >(7)
- B. FUN<int, double >(7) 或 FUN<int>(7)
- C. FUN<int>(7) 或 FUN(7)
- D. FUN(7) 或 FUN<double>(7)

二、填空题

1. 在模板声明中，与 typename 等效的关键字是_____。

2. 在下面的声明中, 函数模板包括(填模板名)_____。

```
template<class T> T neg(T n){ return -n;}
int sub(int m,int n){ return m-n;}
template<typename T>
class OB{
    T d;
public:
    T getData(){ return d;}
};
```

3. 已知一函数模板为

```
template<int n,typename T>
T total(T data[]){
    T s=0;
    for(int i=0;i<=n;i++) s+=data[i];
    return s;
};
```

且数组 a 定义如下:

```
int a[]={1,3,5,7};
```

请用最简略的形式补充完整下面的语句, 使之能够利用上面的模板显示输出数组 a 中所有元素的合计。

```
cout<<_____;
```

4. 在下面程序横线处填上适当字句, 以使该程序执行结果为

```
10
21
33.9
```

```
#include <iostream>
using namespace std;
template <_____>
void min(_____) {
    if(x<y)
        cout<<x<<endl;
    else
        cout<<y<<endl;
}
int main(){
    min(23.5,10);
    min('C',21);
    min(33.9,'X');
    return 0;
}
```

5. 有如下程序:

```
#include<iostream>
using namespace std;
template<class T>
void f(_____) {
```

```

        _____;
    for(int i=0;i<n/2;i++){
        t=a[i],a[i]=a[n-1-i],a[n-1-i]=t;
    }
}
int main(){
    int x[]={6,7,8,9};
    f(x,4);
    for(int i=0;i<4;i++) cout<<a[i]<<' ';
}

```

在上面程序横线处填上适当字句，以使该程序执行结果为：

9 8 7 6

6. 有如下程序：

```

#include<iostream>
using namespace std;

template<typename T>
T Sub(T v1, T v2){ return v1-v2; }

int main(){
    cout<<Sub(5,3)<<', '<<Sub<int>(5.3,3.5);
    return 0;
}

```

程序的输出是_____。

7. 已知

```
int DBL(int n){ return n+n;}
```

和

```
long DBL(long n){ return n+n;}
```

是一个函数模板的两个实例，则该函数模板的定义是_____。

8. 有如下程序：

```

#include<iostream>
using namespace std;
template<class T>
void f(_____) {
    _____;
    for(int i=0;i<n/2;i++){
        t=a[i],a[i]=a[n-1-i],a[n-1-i]=t;
    }
}
int main()
{
    int a[5]={21,22,23,24,25};
    double d[6]={3.1,4.2,5.3,6.4,7.5};
    f(a,5);f(d,6);
    for(int i=0;i<5;i++) cout<<a[i]<<" ";
    cout<<endl;
    for(i=0;i<6;i++) cout<<d[i]<<" ";
}

```

```

    cout<<endl;
    return 0;
}

```

在上面程序横线处填上适当内容，以使该程序执行结果为：

```

25 24 23 22 21
0 7.5 6.4 5.3 4.2 3.1

```

9. 有如下程序：

```

#include <iostream>
using namespace std;

template<class T>
class pair{
public:
    T x,y;
    void show(){
        cout<<"The pair is("<<x <<', '<<y<<')"<<endl;
    }
};

int main(){
    _____;
    c1.show();
    _____;
    c2.show();

    return 0;
}

```

在上面程序横线处填上适当内容，以使该程序执行结果为：

```

The pair is(21,55)
The pair is(21.7,85.4)

```

10. 有如下程序：

```

#include <iostream>
using namespace std;

template<typename T>
T total(int *data){
    T s=0;
    while(*data) s+=*data++;
    return s;
}

int main(){
    int x[]={2,4,6,8,0,12,14,16,18};
    cout<<total(x);
    return 0;
}

```

执行上面程序的输出是_____。

11. 在下面程序横线处填上适当内容，以使该程序执行结果为：5.5。

```

template<typename T>

```

```
T average(T *data) {
    T s=0;
    _____;
    while(data[i]) s+=data[i++];
    return _____;
};

int main(){
    double x[]={2.5,4.5,6.5,8.5,0.0,12.5,14.5,16.5,18.5};
    cout<<average(x);
    return 0;
}
```

12. 有如下的类模板定义:

```
template<typename TYPE>
class MyClass{
    TYPE n;
public:
    MyClass(TYPE k);
    ...
};
```

其中构造函数 MyClass 用 k 的值初始化数据成员 n。因此在模板类体外，构造函数 MyClass 应定义为_____。