1. 以下关于Mac OS X操作系统的说法错误的是()
2. Mac OS X是苹果操作系统软件的Mac OS最新版本
3. Mac OS X包含Darwin和Aqua两大部分
4. Mac OS X 操作系统可以安装在普通的PC机上
5. Mac OS X中的X是罗马数字10,表示版本号
6. 以下关于iOS操作系统的说法错误的是()
7. iOS是苹果公司专门为iPhone开发的操作系统
8. iOS与它基于的Mac OS X操作系统类似,以Darwin为基础
9. iOS是Mac OS 10.5的嵌入式系统版
10. iOS与Mac OS X没有任何关系
11. 以下关于Cocoa的说法错误的是()
12. Cocoa是针对Mac OS X操作系统和iPhone操作系统的一个应用环境
13. Cocoa框架为运行Mac OS X操作系统和iPhone操作系统上的应用程序提供一个运行时环境
14. Cocoa包含Foundation框架,Application Kit框架和UIKit框架
15. Application Kit框架应用于iOS操作系统
16. 以下关于Object-c的说法错误的是()
17. Object-c是苹果Mac OS X上开发的首选语言
18. Object-c是ANSI C的超集,扩展了C语言,使其具备面向对象设计的能力
19. 在Object-c中,不能使用scanf函数进行输入
20. Object-c包含类,消息和继承等面向对象的特征
21. 以下描述正确的是(D)
22. Object-c中,源文件的后缀为.h
23. Object-c中,头文件的后缀为.m
24. Object-c中BOOL类型的值有true和false
25. Object-c和c语言一样,char \*p代表定义了一个字符串
26. 先有如下定义:CGFloat f=10.001; NSLog(@”f=%d”,f);以下描述正确的是
27. 代码最后输出f=10;
28. 编译错误
29. 运行错误
30. 输出f=0;
31. 有如下定义:CGFloat f=10.001; 以下数据类型转正确的是()
32. NSString \*str=[NSString stringWithFormat:@”%d”,f];
33. NSString \*str=[NSString stringWithFormat:@”%ld”,f];
34. NSString \*str=[NSString stringWithFormat:@”%f”,f];
35. NSString \*str=[NSString stringWithFormat:@”%@”,f];
36. 有如下定义:NSString \*str=@”张三”,以下能正确输出的是()
37. NSLog(@”name=”,str);
38. NSLog(@”name=%@”,str);
39. NSLog(”name=%@”,str);
40. NSLog(@”name=”+str);
41. 9. 有如下定义:NSString \*str=@”10.001”,以下能类型转换正确的是()
42. NSInteger f=[str intvalue];
43. NSInteger f=str.intvalue;
44. NSInteger f=[str.intValue];
45. NSInteger f=[str intValue];
46. 开发Mac OS X上的应用程序时,会使用如下框架()
47. Foundation框架和Application Kit框架
48. Foundation框架和UI Kit框架
49. UI Kit框架和Application Kit框架
50. 以上都不正确
51. 以下关于Object-c接口文件的说法错误的是()
52. 接口文件的扩展名为”.h”
53. 接口文件使用”@interface”指令声明
54. 接口文件中的实例变量和实例方法可以定义在一起
55. 接口文件中的实例变量必须定义在一对”{}”括号内
56. 以下关于Object-c实现文件说法错误的是()
57. 实现文件的扩展名为”.m”
58. 实现文件的扩展名为”.c”
59. 实现文件使用”@implementation”指令声明
60. 在实现文件中只实现几口文件中定义的方法

13以下代码能正确为MyRect类的对象myRect分配内存并实例化的是()

1. MyRect \*myRect=[[MyRect alloc]init];
2. MyRect myRect=[[MyRect alloc]init];
3. MyRect \*myRect=[[myRect alloc]init];
4. MyRect myRect=[[myRect alloc]init];

14.以下关于获取函数和设置函数的说法正确的是()

A.获取函数名不能与实例变量同名

B.获取函数名可以与实例变量同名

C 一个类中可以没有获取函数而直接访问实例变量

D以上说法都不正确

15.以下说法正确的是()

A.当创建一个类时,必须为这个类创建两个文件

B.一个类必须拥有两个部分:接口部分和实现部分

C.一个类可以只拥有接口部分

D.一个类可以只拥有实现部分

16.以下定义实例变量的语法正确的是()(选择两项)

A.NSInteger \*width;

B.NSInteger width,height;

C.NSString name;

D.CGFloat salary;

17.以下定义实例方法的语法错误的是()

A.-(NSString)print;

B.-(NSInteger)width;

C.+(NSString \*)print;

D.-(NSInteger \*)width;

18.以下说法正确的是()

A.self类似于java中的this,代表当前调用方法的对象引用

B.self和java中的this用法一致,可以用self.的方式调用当前类的属性和方法

C.在当前类中,调用其他方法时,必须使用self

D.以上说法都不正确

19.以下代码正确的是()(选择两项)

A.id \*name=@”zhangsan”;

B.id name=@”zhangsan”;

C.id \*pool=[NSAutoreleasePool new];

D.id pool=[NSString stringwithFormat:@”%d”,@”1000”];

20.以下说法正确的是()

A.-修饰的方法代表实例方法,可以调用实例变量和静态变量

B.+修饰的方法代表类方法,可以调用类中所有的属性和方法

C.定义方法时,参数名可以和属性名一致,self.property可以调用类中的属性

D以上说法都不正确

21.以下说法正确的是()

A.alloc方法只能为对象创建内存

B.alloc方法只能为对象初始化属性

C.nil和null的意义一样,可以相互替换

D.new关键字可以用于创建对象并初始化

22.以下对init方法说法正确的是()

A.init方法类似于java中的构造方法,用于初始化对象的属性,同样不能有返回类型

B.init方法类似于java中的构造方法,可以为对象创建内存并初始化属性

Cinit方法可以有返回值也可以没有

Dinit方法必须返回当前类的对象.

23.以下关于在Objective-c中实例化对象的说法错误的是（）

1. 使用new方法或者alloc和init方法实例化对象
2. 使用alloc方法分配内存
3. Alloc方法会将类中的各个实例变量的值都设置为零。
4. 调用init方法时不再对实例变量初始化

24.以下代码用于重写父类的init方法，选项中能正确进行初始化的是（）

1. If(self=[super init])

{

}

Retun self;

1. If(self==[super init])

{

}

Retun self;

1. If(self==[super init])

{

}

1. If(self=[super init])

{

}

25.以下关于接口文件的实现文件的说法错误的是（）

1. 在使用Xcode定义类时会自动生成接口文件和实现文件
2. 一个类可以只有一个实现文件
3. 一个类可以有多个接口文件
4. 一个Xcode工程中允许存在多个main函数

26.假设存在以下关于Stud类的定义，则能够使用@property自动标识属性的是（）

@interface Stud:NSObject{

//定义实例变量

NSString \*sid;

NSString \*sname;

}

1. @property Nsstring \*sid,\*sname;
2. @property sid,sname;
3. @property \*sid,\*sname;
4. 以上代码都不对

27.以下关于Objective-C中继承的说法错误的是（）

A.Objective-C可以实现多继承

B.Objective-C 不能实现多继承

C.不能通过继承删除或减少方法

D.super 会从父类的方法列表中开始搜寻，然后调用父类的方法

28.以下关于@class指令的说法错误的有（）

1. @class 指令会包括类的所有信息
2. @class 指令一般用于类的接口文件中
3. 在类的实现文件中，如果要用到引用类中的方法，可以使用@class指令
4. #import 指令可能会导致循环包含编译出错

29.以下关于Objective-C中多态的说法错误的是（）

1. Objective-C中的多态指来自不同类的对象可以定义共享相同名称的方法
2. Objective-C中的多态是借助于静态绑定技术实现的
3. 动态绑定是指在运行时才决定对象所属的类，而并非在编译时决定
4. 动态绑定也称延迟绑定

30.以下关于id数据类型的说法正确的是（）

1. 定义一个变量为id类型时需要加“\*”号
2. id变量中的对象类型在编译时无法确定
3. id变量中的对象类型在运行时无法确定
4. 以上说法都不对

31.对以下代码描述正确的是()

@interface A:NSObject

{

int i,j;

}

@property int i,j;

@end

@interface B:A

{

int z;

}

@property int z;

@end

1. A类中有2个属性,2个方法
2. 编译错误
3. A类中有2个属性,4个方法
4. A类中只有2个属性,没有方法

32.对以下代码描述正确的是()

@interface A:NSObject

{

int i,j;

}

@property int i,j;

@end

@interface B:A

{

int z;

}

@property int z;

@end

A.B类中有1个属性,1个方法

B.编译错误

C.B类中有3个属性,没有方法

D.B类中只有3个属性,6个方法

33.对以下代码描述正确的是()

@interface A:NSObject

{

int i,j;

}

@property int i,j;

@end

@interface B:A

{

}

@end

A.B类中有1个属性,1个方法

B.编译错误

C.A类中没有属性,也没有方法

D.A类中只有2个属性,4个方法

34.以下说法正确的是

A.方法的重写是针对的同一个类中的方法

B.方法的重写是指子类必须要重写父类的方法

C.方法的重写是指子类可以扩展父类的同名方法,重写时只要名字相同,返回类型和参数都可以不同

D以上说法都不正确

35.以下对main函数说法正确的是

A.一个项目中,main方法可写可不写

B.main方法返回一个整型数值,且值必须为0或1,代表成功或失败

C.main方法中必须带有2个参数,一个是整型,一个是字符型

D以上说法都不正确

36.Object-C中语句以()结尾

A.换行符

B..

C.:

D.;

37.以下不是Object-c中的数据类型的是()

A.char

B.unsigned int

C.long double

D.short

38.对于继承说法正确的是()

A.子类可以继承父类的所有属性和方法

B.子类可以任意访问父类中的属性

C.对子类修改时,不会对父类造成任何影响

D.子类只能增加实例变量和方法

39.判断两个字符串的值是否相等,以下代码正确的是()

A.str1==str2

B.str1 isEqualToString str2

c.[str1 isEqualToString str2]

D.str1.isEqualToString(str2)

40.执行for(i=0;i<10;++i);后,i的值为()

A.10

B.11

C.0

D.9