

2010~2011 学年第一学期期末考试试卷

《软件技术基础 2》(C++64 学时) (A 卷)

(考试时间 2011 年 1 月 12 日)

一. 选择题 (共 20 题, 每小题 1 分)

1. 定义变量: int a; float b; double c; char d='c'; 则表达式 a*(b+c)/d 的数据类型为 ()

- A) int B) double C) float D) char

2. 下列函数原型声明中, 错误的是 ()

- A) int function (int m, int n); B) int function (int, int)
C) int function (int m=3, int n) D) int function (int &m, int &n)

3. 若 a、b、x、y 均为整型变量, 以下表达式中符合 C++ 语法的是 ()

- A) (a*b)++ B) a%4.0 C) a=1/2*(x=y=10, x*3) D) a=3.5

4. 若 a=1, b=15, c=0, 则 (a>b)>c 的值为 ()

- A) 1 B) 0 C) 非 0 D) 该表达式不符合 C++ 语法

5. 有定义语句: int a[]={1,2,3,4,5,6,7}, *p=a, *q=&a[5]; 则下列表达式中值不等于 5 的是

- A) *(--p) B) *(--q) C) --q D) q-p

6. 下列数组定义中, 错误的是 ()

- A) int a; int *p=&a B) char a[]="c++"; C) const int N=5; int a[N+5] D) int n=5; int a[n]

7. 以下指针用法错误的是 ()

- A) int a; int *p=&a; B) int a=0, b=1; int *p=&a, *p=b;

- C) int a[3], *p=&a; *p=3; D) int a[3], *p=a; *p=3;

8. 已知 int x, a=0, b=2; 语句 x=a++&&b++||++b; 执行后, x, a, b 的值为 ()

- A) 1, 1, 3 B) 1, 1, 4 C) 0, 0, 2 D) 0, 1, 2

9. 下列语句中正确的是 ()

- A) int a[]="string"; B) int a[4]={0,1,2,3,4};

- C) char a="string"; D) int a[10]={0};

10. 下列函数中, 可以提高程序执行效率的是 ()

- A) 递归函数 B) 内联函数 C) 系统函数 D) 函数模板

11. 定义一个函数实现交换 x 和 y 的值, 并使实参的值也交换, 能够实现此功能的是 ()

A) void swap (int x, int y) {int t; t=x; x=y; y=t}

B) void swap (int *x, int *y) {int t; t=x; x=y; y=t}

C) void swap (int *x, int *y) {int t; t=*x; *x=*y; *y=t}

D) void swap (int *x, int *y) {int *t; t=x; x=y; y=t}

12. 若有 fun() {static int x;}, 下列关于变量 x 的说法, 错误的是 ()

- A) x 的初始值为 0 B) x 是全局变量

- C) 函数 fun 执行结束时 x 的值保留 D) x 具有静态生存期

13. 下列关于 main 函数说法中, 正确的是 ()

- A) 每个程序可有多个 main 函数 B) main 函数一定没有返回值

- C) main 函数的返回值类型缺省为 int D) 每个 main 函数一定有一个返回值

14. 下面关于类的说法中, 正确的是 ()

- A) 通过私有派生, 可以在派生类中直接访问基类的私有成员

- B) 不能定义抽象类的对象和引用, 但可以定义抽象类的指针

P=地址
*P=内容

- C) 一个类的友元函数可以访问该类对象的任何数据成员
D) 类的私有成员和公有成员的说明有严格的顺序要求
15. 下列关于构造函数的说法中，错误的是（ ）
A) void 不能作为构造函数的类型 B) 构造函数是在创建对象时自动调用的
C) 构造函数不能重载 D) 构造函数的函数名必须与类同名
16. 在继承环境中，动态多态性由（ ）实现
A) 友元函数 B) 重载函数 C) 虚函数 D) 静态成员函数
17. 若在表达式 y/x 中，“/”是作为成员函数重载的运算符，则该表达式可解释为（ ）
A) $x.\text{operator } / (y)$ B) $\text{operator } / (x,y)$ C) $y.\text{operator } / (x)$ D) $\text{operator } / (y,x)$
18. 若有说明： int a=4, fstream aa("bb.txt"); 则可以将 a 的值写入文件 bb.txt 中的是（ ）
A) aa.put(a) B) aa<<a C) cout<<a D) aa.write(a,szeof(int))
19. 下面的说法，正确的是（ ）
A) 顺序存储只能实现线性表的存储，不能实现二叉树的存储
B) 一棵二叉树一定是一棵树
C) 栈是先进先出的数据结构，队列是后进先出的数据结构
D) 满二叉树一定是完全二叉树
20. 下列说法中，正确的是（ ）
A) 在顺序表中进行查找，二分查找算法是最快的
B) 插入、删除操作频繁的线性表适合用顺序表
C) 使用带头节点的链表的主要原因是方便记录链表中的元素个数
D) 采用顺序循环队列可以解决顺序队列的假溢出问题

二. 写出以下程序的运行结果（共 24 分，每小题 4 分）

```
1. #include<iostream.h>
void main(){
    char s[80];
    cin>>s;
    for(int i=0;s[i];i++){
        switch(s[i]){
            case 'L':cout<<"向左走！";break;
            case 'R':cout<<"向右走！";break;
            case 'f':cout<<"向前走！";break;
            case 'B':;
            case 'b':cout<<"向后走！";break;
            case 'S':;
            case 's':cout<<"停止！";break;
            default: cout<<"原地不动！";break;
        }
    }
}
```

如果从键盘输入： qrLbFs ， 则输出结果是：

```
2. #include<iostream.h>
const int SIZE=6;
void func(int *p,int &a){
    static int b;
    b+=*(p-1);
    a+=b;
}
void main(){
    int a[SIZE]={5,6,9},i;
    for(i=3;i<SIZE;i++)
        func(a+i,a[i]);
    for(i=0; i<SIZE;i++)
        cout<<a[i]<<" ";
}
```

输出结果:

```
3. #include<iostream.h>
const int B=2;
void main(){
    int p=1,s=1;
    while(s<50) {
        p*=B;
        s+=p;
    }
    cout<<"p="<<p<<"s="<<s<<endl;
}
```

输出结果:

```
4. #include<iostream.h>
#include<math.h>
void conver(int);
void main(){
    int num;
    cout<<"Input a integer:"<<endl;
    cin>>num;
    if(num<0)
        cout<<"-";
    conver(num);
    cout<<endl;
}

void conver(int n){
    if(n>=0&&n<=9)
```

```

        cout<<n;
    else if(n>=9&&n<=1)
        cout<<-n
    else{
        cout<<abs(n%10); //abs 为求绝对值的系统函数
        conver(n/10); }

}

```

若输入-789，则输出结果是：

```

5. #include<iostream.h>
class goods{
    static int totalWeight;
    int weight;
public:
    goods(int w){
        weight=w;
        totalWeight+=w;}
    goods(goods& gd){
        weight=gd.weight;
        totalWeight+=weight;}
    ~goods(){
        totalWeight-=weight;}
    int getwg(){
        return weight;}
    static int getTotal(){
        return totalWeight;}
};

int goods::totalWeight=0;
void main(){
    int w;
    cout<<"The initial weight of goods:"<<goods::getTotal()<<endl;
    w=38;
    goods g1(w);
    w=59;
    goods g2(w);
    cout<<"The total weight of goods:"<<goods:: getTotal()<<endl;
}

```

输出结果：

```

6.#include<iostream,h>
class Base{
private:
    int a;
public:

```

```

Base(){a=10;cout<<"基类缺省构造函数\n";}
Base(int i){a=i;cout<<"基类缺带参构造函数\n";}
~Base(){cout<<"基类析构函数\n";}
virtual void print(){cout<<"a="<<a<<endl;}
};

class Derived:public Base{
private:
    int b;
public
    Derived(){b=20;cout<<"派生类缺省构造函数\n";}
    Derived(int i,int j):Base(i),b(j){cout<<"派生类带参构造函数\n";}
    ~Derived(){cout<<"派生类析构函数\n";}
    void print(){Base::print();cout<<"b="<<b<<endl;}
};
void main(){
    Base *pa,b;
    Derived d(1,2);
    pa=&b;
    pa->print();
    pa=&d;
    pa->print();
}
输出结果:

```

三、程序填空 (共 32 分, 每空 2 分)

1.下面程序的功能是将键盘上输入的一串字符串存入一个名为 data.txt 的文件。字符串中包含空格并以“#”作为输入结束的标志。

```

#include<iostream.h>
#include<fstream.h>
#include<string.h>
void main()
{
    char s2[80],ch;
    int i=0;
    cout<<"输入一个字符串: "<<endl;
    do{
        _____;
        s2[i]=ch;
        i++;
    }while(_____);
    s2[i]=0;
    _____;
    outfile<<s2<<endl;
}

```

2.重写教材中单链表类 Chain 的插入函数, 函数原型为 void Insert(const char&x); 使用该函数

完成插入后，线性表中元素按递减顺序排列。

```
void Chain::Insertasc(const char &x){  
    Node *p=head;  
    while(_____  
        p=p->next;  
    Node*q;  
    q=_____;  
    q->data=x;  
    _____;  
    p->next=q;  
    length++;  
}
```

3. 函数 int palindrome(int num) 的功能是判断一个数是否是回文数（所谓回文数，就是从左至右读和从右至左读是同一个数，如 101）。若是回文数，函数则返回 1，否则返回 0。在 11~999 之间寻找这样的数，它不仅是回文数，而且它的平方和立方都是回文数。

```
#include<iostream.h>  
int palindrome(int num);  
void main(){  
    for(int j=11;j<=999;j++)  
        if(palindrome(j))  
            cout<<"回文数："<<j<<endl;  
}  
  
int palindrome(int num){  
    int temp=num,s=0;  
    while (temp!=0){  
        s=s*10+_____;  
        temp/=10;  
    }  
    if(_____  
        return 1;  
    else  
        return 0;  
}
```

4. 下面程序从键盘将 10 名学生的学号和成绩输入到对象数组，按成绩由高到低进行排序后，输出排名前 30% 的学生。

```
#include<iostream.h>  
#include<string.h>  
class Student{  
    char sNo[10];  
    int score;  
public:  
    Student(char no[],int g){  
        strcpy(sNo,no);  
        score=g;
```

```

    };
    Student();
    void out();
    friend void sort(Student *s, int n);
};

out(){
    cout<<"学号"<<sNo<<"成绩"<<score<<endl;
};

void sort(Student *s,int n){
    Student temp;
    int i,j,k;
    for(i=0;i<n-1;i++){
        k=i;
        for(j=i+1;j<n;j++)
            if(
                k=j;
                temp=s[i];
                s[i]=s[k];
                s[k]=temp;
            }
    }
}

void main(){
    const int number=10;
    Student teststudent[number];
    int i,s;
    char no[10];
    for(i=0;i<number;i++){
        cin>>no>>s;
        teststudent[i]=Student(no,s);
    }
    sort(teststudent,number);
    for(i=0;_____,i++)
        teststudent[i].out();
}

```

5.下列程序中，函数 fun 的功能是：计算 x 所指数组中 N 个数的平均值（规定所有数均为正数），平均值通过形参返回主函数，并将小于平均值且最接近平均值的数作为函数值返回。在主函数中输入十个正整数，调用该函数计算小于平均值且最接近平均值的数并输出到屏幕。

例如，有 10 个正数：46 30 32 40 6 17 45 15 48 26，平均值为：30.5

主函数中输出：The close number is:30

```

#include<iostream.h>
#define N 10
double fun(double x[],double *av)
{

```

```

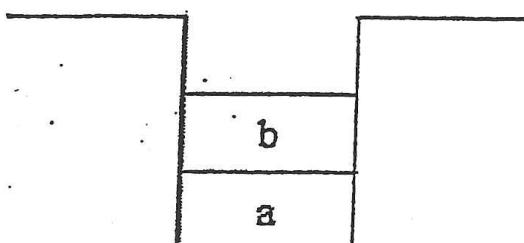
int i,j;
double d,s;
s=0;
for(i=0;i<N;i++)
    s=s+x[i];
    _____=s/N;
d=32767;      //设定差值上界
for (i=0;i<N;i++)
    if((x[i]<*av)&&(*av-x[i]<=d))
    {
        d=*av-x[i];
        j=_____;
    }
    return _____
}
void main()
{
    int i;
    double x[N],av,m;
    for(i=0;i<N;i++)
        cin>>x[i];
        cout<<x[i]<<"";
    cout<<endl;
    m=fun(_____);
    cout<<"\nThe average is:\n"<<av<<endl;
    cout<<"The close number is:\n"<<m<<endl;
}

```

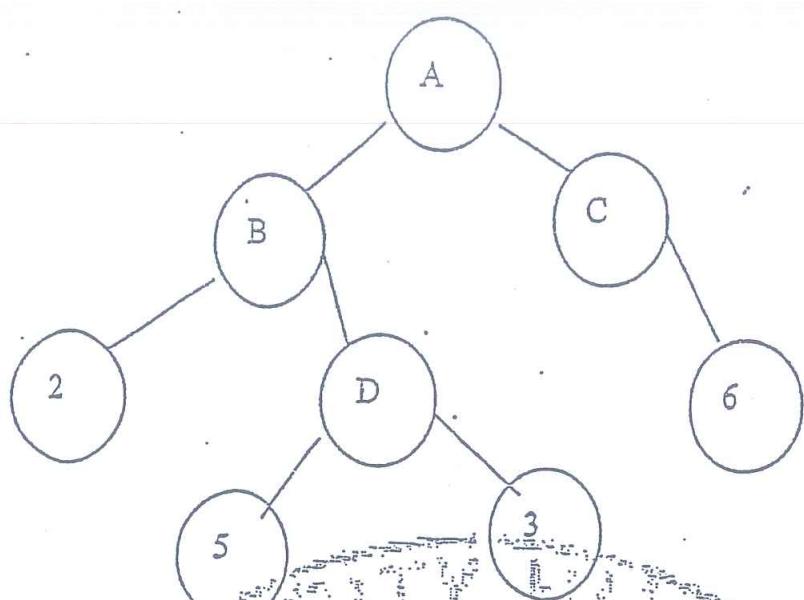
四. 简答题

1. 假设栈的容量足够大，栈状态如下图所示。写出所有可能的出栈序列。

d c



2. 计算下面扩充二叉树的 WPL 值，画出对应的哈夫曼树



3. 如果哈希函数为 $k \% 7$ ，采用线性探查序列处理冲突。画出数据集合 {14, 37, 45, 76, 93, 69} 对应的哈希表。

4. 给定待排序列 {15, 14, 21, 11, 33, 28, 35, }，采用冒泡排序算法按升序排序。请写出每趟排序后的结果。

五. 编程题

编写函数 `int maxi(int *s, int n)`，其功能是找出 n 个整数中的最大数并返回该数。编写主要程序，利用该函数找出 5×5 的二维数组中每行中最大的数，并在屏幕输出，每个数占一行。二维数组的数据在主程序中由键盘输入。

C++(2011.1)答案

一. 选择题

题号	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
答 案	B	C	C	B	A	D	C	A	D	B	C	B	C	C	C	C	C	B	D	D

二. 阅读程序写结果

1. 原地不动！向右走！向左走！向后走！向前走！停止！

2. 5 6 9 9 18 36

3. P=32 s=63

4. -987

5. The initial weight of goods:0

The total weight of goods:97

6. 基类缺省构造函数

基类带参构造函数

派生类带参构造函数

a=10

a=1

b=2

派生类析构函数

基类析构函数

基类析构函数

三. 程序填空

1. cin.get(ch)或 cin>>ch 或 ch=cin.get() ch='*' ofstream outfile("data.txt")

2. p->next!=NULL&&p->next->data>x new Node q->next=p->next

3. &&palindrome(j*j)&& palindrome(j*j*j) temp%10 s.num

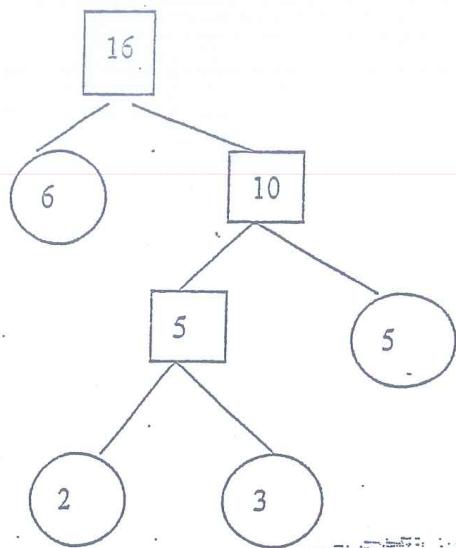
4. void Student:: s[k].score<s[j].score i<number*0.3

5. *av i x[j](或 x[i]; j) x,&av

四. 简答题

1. dcba, cdba, cbda, cbad, bcda, bcad, bdca, bacd, badc

2. WPL=2*2+5*3+3*3+6*2=40



3.

0	1	2	3	4	5	6
14	69	37	45	93		76

4. 第一趟: 14, 15, 11, 21, 28, 33, 35

第二趟: 14, 11, 15, 21, 28, 33, 35

第三趟: 11, 14, 15, 21, 28, 33, 35

五. 编程题

```
#include <iostream.h>
int max(int *s,int n);
void main()
{
    int a[5][5],max,ij;
    cout << "请输入 5 行 5 列数组的数据: \n";
    for(i=0;i<5;i++)
        for(j=0;j<5;j++)
            cin >> a[i][j];
    for(i=0;i<5;i++)
    {
        rmax=max(a[i],5);
        cout << "第" << i << "行的最大数:" << rmax << endl;
    }
}
int max{int *s,int n}
{
    int max=s[0];
    for(int i=1;i<n;i++)
        if(max<s[i])
            max=s[i];
    return max;
}
```

//程序框架对给 2 分
 //输入部分对给 2 分
 //1分
 //1分
 //输出对 1分
 //1分
 //1分
 //2分
 //1分
 //1分