2024-2025高级语言程序设计回忆版

1. **程序阅读题（5题共30分）**

以下代码回忆来自*Kilig*

**1.**

#include <iostream>

using namespace std;

int main()

{

int ar[]={2,2,2,3,4,4,5,6,6,6,6,7,8,8,9,10,10,10};

int n =sizeof(ar)/sizeof(int);

for(int i=0;i<n-1;i++)

{

for(int j=i+1;j<n;j++)

{

if(ar[i]==ar[j])

{

cout<<ar[i]<<' ';

for(int k=j+1;k<n;k++)

{

ar[k-1]=ar[k];

}

n--;

j--;

}

}

}

cout<<endl;

for(int i =0;i<n;i++)cout<<ar[i]<<' ';

cout<<endl;

return 0;

}

**2**

#include <iostream>

using namespace std;

int main()

{

char s[10]="student",c;

int n=7;

for(int i =0;i<n/2;i++)

{

c=s[i];

s[i]=s[n-i-1];

s[n-i-1]=c;

}

cout<<s<<endl;

return 0;

}

**3**

#include <iostream>

using namespace std;

void func()

{

static int cnt=0;

cnt++;

int k=3;

switch(cnt)

{

case 1:k++;

cout<<k<<endl;

break;

case 2:k+=cnt;

cout<<k<<' ';

case 3:k+=3;

cout<<(k+cnt)<<endl;

break;

default:

cout<<(k+cnt)%5<<endl;

}

}

int main()

{

for(int i =0;i<5;i++)func();

return 0;

}

**4**

#include <iostream>

using namespace std;

void func(int \*a,int &n)

{

n--;

for(int i =0;i<n;i++,n--)

{

a[n]+=a[i];

}

}

int main()

{

int a[8]={1,2,3,4,5,6,7};

int m=8;

func(a,m);

for(int i =0;i<m;i++)cout<<a[i]<<" ";//这里是8还是m忘了

cout<<endl<<m<<endl;

return 0;

}

**5**.已知：

struct Node

{

int data;

Node \*next;

};

struct lst

{

Node \*head;

int length;

};

void add\_node(lst \*l,Node \*p)

{

static int c=0;

if(l->head==NULL)

{

l->head=p;

c=p->data;

}

else

{

if(c>p->data)

{

p->next=l->head;

l->head=p;

}

else

{

Node \*h=l->head;

while(h->next)h=h->next;

h->next=p;

}

}

l->length++;

}

给出8个数字，插入这8个数字，请写出链表中8个数字的顺序

**其他群u回忆如下：**

写的贼丑的unique，先输出删除的数，后输出剩下的数

字符串倒序（”student ”）

fun() 处理数组 ，主要考察静态局部变量和swtich，注意没有break和输出空格还是换行

链表操作，高于key放末尾否则放头部

1. **编程题（2题共20分）**

1.

输出到prime.txt里。输出2到N的素数，如果N为偶数，输出2\*C个；如果N为奇数，输出2\*c-1个。如果超过N，输出所有素数

切割字符串

字符串切割成 1个字符，两个字符，三个字符....

**三、综合设计题（2题共50分）**

myvector类

重载[]

friend 重载＝

拷贝构造函数

Account类

创造一个指向基类的指针，调用三个函数

{protected:

double balance;

public:

virtual void display()const;

virtual void deposit(double);

virtual void withdraw(double);

//虚析构函数

//构造函数

}

checkingaccount和savingsaccount两个（虚函数多态派生继承）

saving那个有interestRate

checking那个有transaction