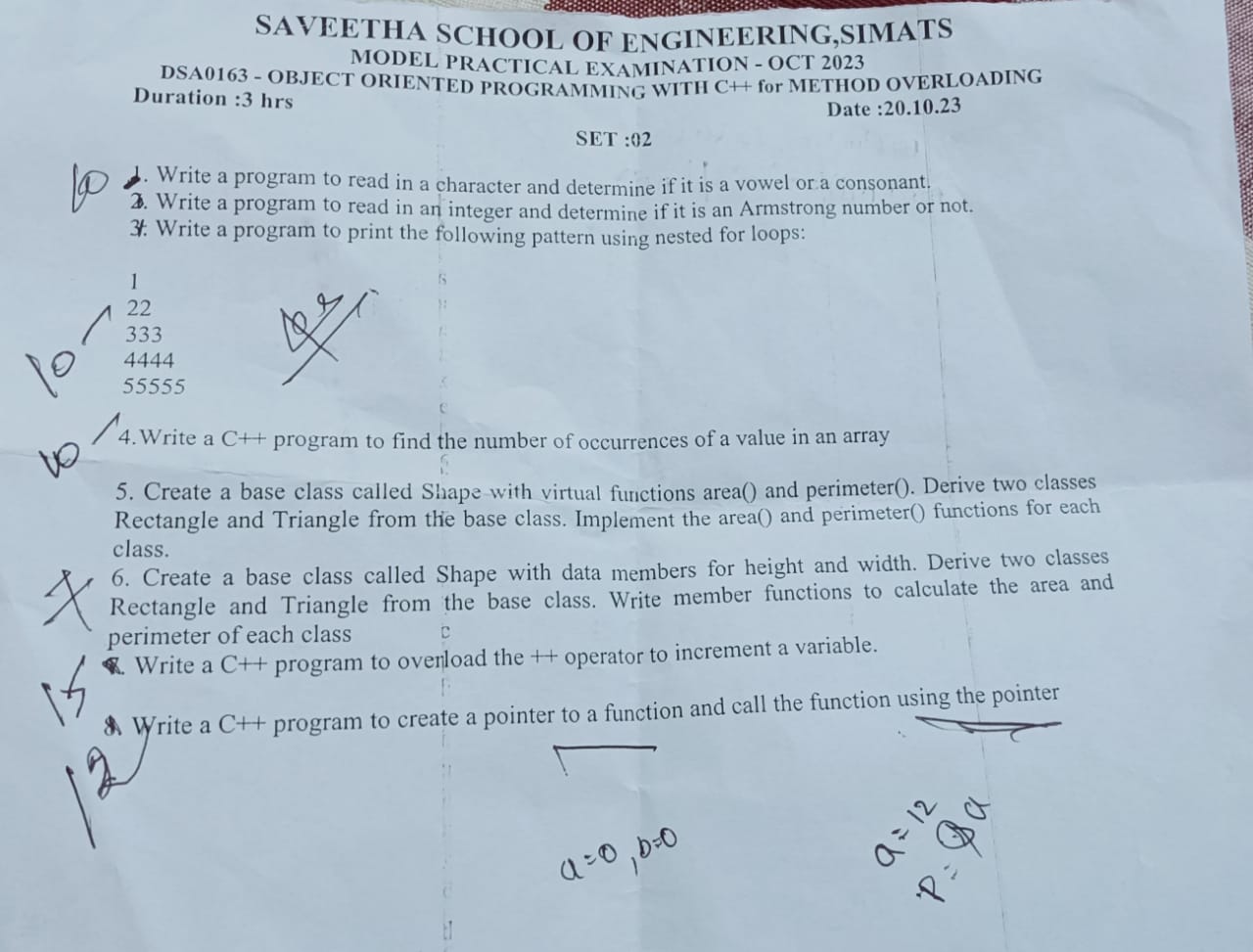
SAVEETHA SCHOOL OF ENGINEERING,SIMATS MODEL PRACTICAL EXAMINATION-OCT 2023

DSA0163-OBJECT ORIENTED PROGRAMMING WITH C++ for METHOD OVERLOADING

QUESTION PAPER:



1.VOWELS and CONSONANTS?

CODE:

#include<iostream>

using namespace std;

int main(){

int a=0,b=0,c=0;

char n[50];

cin>>n;

int l=strlen(n);

for(int i=0;i<l;i++){

if(n[i]=='a'||n[i]=='e'||n[i]=='i'||n[i]=='o'||n[i]=='u'||n[i]=='A'||n[i]=='E'||n[i]=='I'||n[i]=='O'||n[i]=='U')

{

a++;

}

else

{

c++;

}

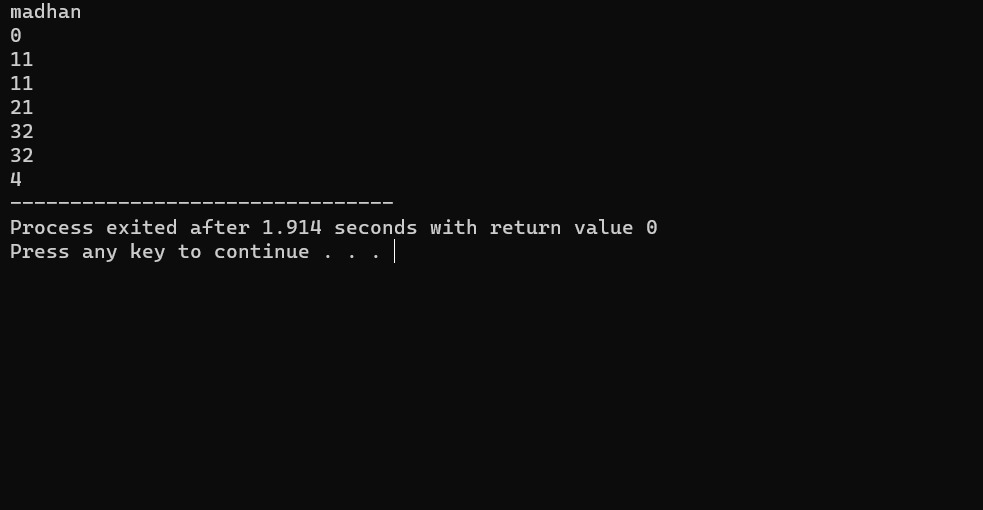
cout<<a<<endl;

cout<<c;

}

}

OUTPUT:



2.ARMSTRONG?

CODE:

#include<iostream>

using namespace std;

int main(){

int a=0,b,c=0,n,d,e;

cout<<"enter integers:";

cin>>n;

d=n;

e=n;

while(n!=0){

n=n%10;

a++;

}

while(d!=0){

b=d%10;

c=c+(pow(b,a));

d=d/10;

}

if(c==e)

cout<<"armstrong"<<endl;

else

cout<<"not armstrong";

return 0;

}

3.PATTERN?

CODE:

#include<iostream>

using namespace std;

int main(){

int rows=5;

for(int i=1;i<=rows;i++){

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

cout<<i<<" ";

}

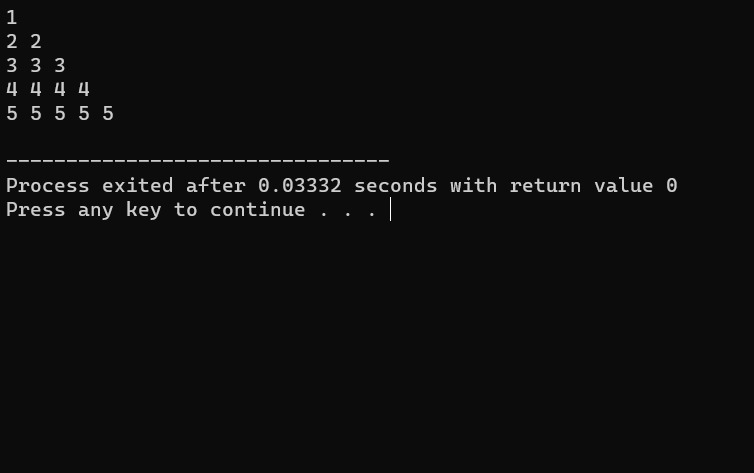
cout<<endl;

}

return 0;

}

OUTPUT:



4.OCCURRENCES?

CODE:

#include<iostream>

using namespace std;

int countoccurrences (int arr[],int size,int value){

int count=0;

for(int i=0;i<size;i++){

if(arr[i]==value){

count++;

}

}

return count;

}

int main(){

int arr[]={1,2,2,3,4,2,5};

int size=sizeof (arr)/sizeof (arr[0]);

int value=2;

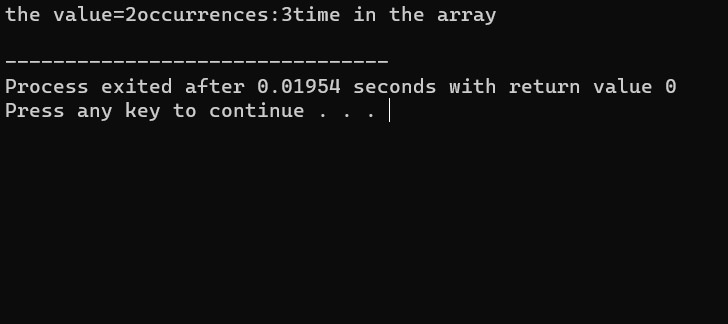
int result=countoccurrences(arr,size,value);

cout<<"the value="<<value << "occurrences:"<<result<<"time in the array"<<endl;

return 0;

}

OUTPUT:



6.SHAPE?

CODE:

#include<iostream>

using namespace std;

class shape

{

public:

int a,b;

void get()

{

cout<<"enter height:"<<endl;

cin>>a;

cout<<"enter width:";

cin>>b;

}

void display()

{

cout<<a;

cout<<b;

}

};

class rectangle:public shape

{

public:

void set(){

cout<<"enter height:"<<endl;

cin>>a;

cout<<"enter width:";

cin>>b;

}

void disp()

{

cout<<a;

cout<<b;

return (a\*b);

}

};

class triangle:public shape

{

public:

void go(){

cout<<"enter height:"<<endl;

cin>>a;

cout<<"enter width:";

cin>>b;

}

void dis()

{

cout<<a;

cout<<b;

return 1/2\*a\*b;

}

};

int main(){

rectangle q;

q.set();

q.disp();

triangle w;

w.go();

w.dis();

}

7.OPERATOR OVERLOADING?

CODE:

#include<iostream>

using namespace std;

class A

{

public:

int a;

void operator ++(){

++a;

}

};

int main(){

A A1;

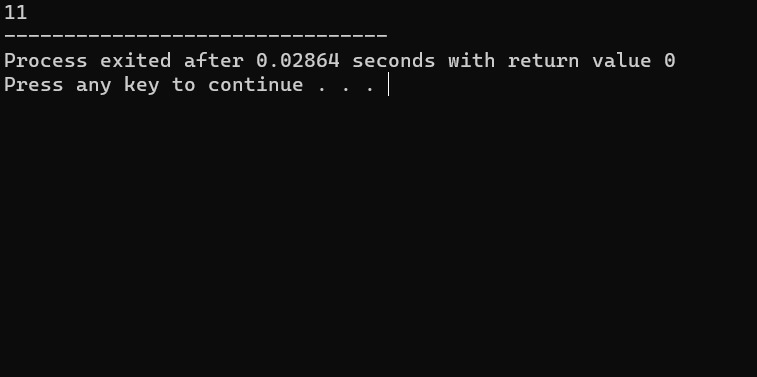
A1.a=10;

++A1;

cout<<A1.a;

}

OUTPUT:



8.POINTER?

CODE:

#include<iostream>

using namespace std;

void red(int\*n){

cout<<\*n;

}

int main(){

int a;

a=12;

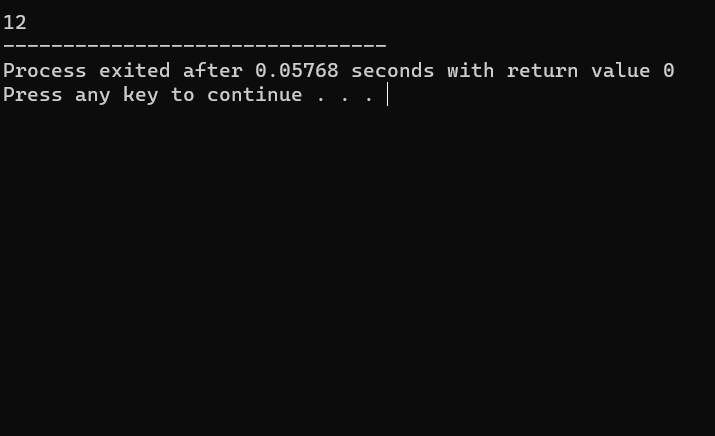
int\*p;

p=&a;

red(p);

}

OUTPUT:



TOTAL =64/100.