Aşağıdaki işi yapan C++ programı için gerekli kodları tamamlayınız. Ekranda en üstte “Küre için Yüzey Alanı ve Hacim Hesabı” ve “Pi Sayısı : 3.141593 Alınacak” satırları görünsün. Yarıçap “Kürenin Yarıçapı : ” yazısı ile kullanıcıdan istensin. Yarıçap reel sayı ise kürenin yüzey alanı ve hacmi hesaplanıp anlaşılır biçimde ekranda yazdırılsın. Yarıçap reel sayı değilse veya reel sayı olup 0 değilse tekrar istensin. Döngüden çıkılınca programı sonlandırmak için bir tuşa basmak gereksin.

#include <iomanip>

#include <iostream>

using namespace std;

#define pi 3.141593

void main() {

cout << "\tKure icin Yuzey Alani ve Hacim Hesabi\n\n";

cout << setw(25) << "Pi Sayisi : " << pi << " Alınacak\n\n";

double r, yalan, hacim;

while(true) {

cout << setw(25) << "Kurenin Yaricapi : ";

cin >> r;

if (cin.good()) {

yalan = 4 \* pi \* r \* r;

hacim = yalan \* r / 3;

cout << endl << fixed << setw(25);

cout << "Kurenin Alani : " << yalan;

cout << endl << setw(25);

cout << "Kurenin Hacmi : " << hacim << "\n\n";

if (r == 0) break;

} else cin.clear();

fflush(stdin);

}

system("PAUSE");

}

Aşağıdaki işi yapan CPP programı için gerekli kodları yazınız. “Name = ” yazısı ekrana gelerek name değişkeni için bir girdi isteniyor. Name değişkeninin kapasitesi maksimum 10 karakter olduğundan girdinin fazlalık kısmı alınmıyor. Name değişkeninin içeriği “Name : ” yazısından, uzunluğu “Uzunluk : ” yazısından sonra ekrana yazdırılsın. Bu işlemler en son boş satır ile beraber 4 satırda gerçekleşsin ve name değişkeni “htemurtas” olana kadar tekrarlansın. Bitiminde programı sonlandırmak için bir tuşa basmak gereksin.

#include <iostream>

using namespace std;

void main() {

const int n = 11;

char name[n];

while (true)

{ cout << " Name = ";

cin.get(name, n);

//cin.getline(name, n); cin.clear();

fflush(stdin);

cout << " Name : " << name << endl;

cout << "Length : " << strlen(name) << "\n\n";

if (!strcmp(name, "htemurtas")) break;

}

system("PAUSE");

}

Aşağıdaki işi yapan C++ programı için gerekli kodları tamamlayınız. Ekranın en üstünde “İkinci Dereceden Polinomun Köklerinin Hesaplanması” satırı görünsün. Polinomun katsayıları ( a, b, c ) sırasıyla kullanıcıdan istensin. Katsayıların reel sayı girildiği kabul edilsin. Köklerin olup olmama durumuna göre ekranda “Kökler Sanal”, “Kökler Reel” ve “Kökler Katlı” yazılarından biri görünsün. Kökler varsa hesaplanıp ekranda yazdırılsın. Bitiminde programı sonlandırmak için bir tuşa basmak gereksin.

#include <iostream>

using namespace std;

void main()

{

double a, b, c, x1, x2, delta;

cout << "İkinci Dereceden Polinomun Köklerinin Hesaplanması\n";

cout << "\ta = ";

cin >> a;

cout << "\tb = ";

cin >> b;

cout << "\tc = ";

cin >> c;

delta = b\*b - 4 \* a\*c;

if (delta < 0) cout << "Kokler Sanal\n\n";

else {

if (delta == 0) cout << "Kokler Katlı\n";

else cout << "Kokler Reel\n";

x1 = (-b - sqrt(delta)) / (2 \* a);

x2 = (-b + sqrt(delta)) / (2 \* a);

cout << " x1 = " << x1 << endl;

cout << " x2 = " << x2 << "\n\n";

}

system("PAUSE");

}

Aşağıdaki işi yapan CPP programı için gerekli kodları tamamlayınız. Eleman sayısı 5 ile 15 arasında rastgele atanan bir tamsayı dizisi tanımlansın. “Dizi Elemanları : ” yazısından sonra dizinin her bir elemanı -50 ile +50 arasında rastgele atanıp sırasıyla aralarına virgül konularak ekrana yazdırılsın. “Eleman Sayısı : ” yazısından sonra dizinin eleman sayısı ekrana yazdırılsın. Dizinin en küçük ve en büyük eleman değerleri bulunup anlaşılır biçimde ekrana yazdırılsın. İşlemler bitince programı sonlandırmak için bir tuşa basmak gereksin.

#include <iostream>

#include <iomanip>

using namespace std;

void main()

{

srand(time(NULL));

int i, min, max;

int n = 5 + rand() % 11;

int \*a = new int[n];

cout << "Dizi Elemanları : ";

for (i = 0; i < n; i++) {

a[i] = rand() % 101 - 50;

if (i != 0) cout << ", ";

cout << a[i];

}

min = max = a[0];

for (i = 1; i < n; i++) {

if (a[i] < min) min = a[i];

if (a[i] > max) max = a[i];

}

cout <<"\n\nEleman Sayisi : " << n;

cout <<"\n\nMin Sayi : " << setw(3) << min;

cout <<"\nMax Sayi : " << setw(3) << max << "\n\n";

system("PAUSE");

}

Aşağıdaki işi yapan CPP programı için gerekli kodları yazınız. Önce, 10 elemanlı bir tamsayı dizisi tanımlansın. Dizinin her bir elemanı 0 ile 999 arasında rastgele atansın ve sırasıyla ekranda gösterilsin. Sonra, dizi küçükten büyüğe doğru sıralansın ve tekrar ekranda gösterilsin. Her dizi gösterimi sonrası boş bir satır eklensin. İşlemler bitince programı sonlandırmak için bir tuşa basmak gereksin.

#include <iomanip>

#include <iostream>

using namespace std;

void main()

{

bool drm;

srand(time(NULL));

const int n = 10;

int i, j, a[n];

for (i = 0; i < n; i++)

{ a[i] = rand() % 1000;

cout << setw(5) << a[i];

}

cout << endl << endl;

do

{ drm = false;

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

{ if (a[i] > a[i+1])

{ j = a[i];

a[i] = a[i+1];

a[i+1] = j;

drm = true;

}

}

} while (drm);

for (i = 0; i < n; i++)

cout << setw(5) << a[i];

cout << endl << endl;

system("PAUSE");

}

Aşağıdaki işi yapan CPP programı için gerekli kodları yazınız. Bekletme süresini dışarıdan ms olarak alan “Beklet” altprogramı oluşturulsun ve program içinde kullanılsın. “İleri Say : ” yazısından sonra her 250 ms zaman aralığında bir tamsayı olmak üzere sırasıyla 1 den 15’e kadar olan tamsayılar ekranda görünsün. Tamsayılar arasında virgül ve boşluk olsun. Bu işlemler hem for döngüsü ile hem de while döngüsü ile olmak üzere iki kez yapılsın. Her gösterim sonrası boş bir satır eklensin. İşlemler bitince programı sonlandırmak için bir tuşa basmak gereksin.

#include <ctime>

#include <iostream>

using namespace std;

void beklet(clock\_t sure)

{

clock\_t t = sure + clock();

while (t > clock());

}

void main()

{

int i, n = 15;

cout << "Ileri Say : ";

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

{ if (i > 1) cout << ", ";

cout << i;

beklet(250);

}

printf("\n\nIleri Say : ");

i = 0;

while (i < n)

{ if (i > 0) cout << ", ";

cout << ++i;

beklet(250);

}

printf("\n\n");

system("PAUSE");

}

Aşağıdaki işi yapan CPP programı için gerekli kodları yazınız. 18 satır ve 12 sütundan oluşan iki boyutlu bir tamsayı matrisi tanımlansın. Matrisin her bir elemanı -100 ile +100 arasında rasgele atansın. Satır ve sütun sayıları hem dosyaya hem de ekrana yazdırılsın. Bir alt satıra geçilip matrisin her bir elemanı matris formunda hem dosyaya hem de ekrana yazdırılsın. Sonra, dosya kapatılsın ve ekrana "A Matrisi Dosyaya Kaydedildi" yazısı gelsin. Dosya ve ekrandaki yazılar anlaşılır biçimde olsun. Dosya ismi “file.txt” olsun. İşlemler bitince programı sonlandırmak için bir tuşa basmak gereksin.

#include <iomanip>

#include <fstream>

#include <iostream>

using namespace std;

void main()

{

srand(time(NULL));

int m, n;

const int M = 18, N = 12;

int a[M][N];

ofstream fw;

fw.open("file.txt");

fw << setw(6) << M << setw(6) << N << endl;

cout << setw(6) << M << setw(6) << N << endl;

for (m = 0; m < M; m++)

{ for (n = 0; n < N; n++)

{ a[m][n] = -100 + rand() % 201;

fw << setw(6) << a[m][n];

cout << setw(6) << a[m][n];

}

fw << endl;

cout << endl;

}

fw.close();

cout << "A Matrisi Dosyaya Kaydedildi\n\n";

system("PAUSE");

}

Aşağıdaki işi yapan CPP programı için gerekli kodları tamamlayınız. İsmi “file.txt” olan dosya bir tamsayı matrisinin bilgilerini tutmaktadır. Bu dosya okunmak için açılsın. Açılamazsa ekrana “Dosya Açılamadı” yazsın. Dosya açılırsa önce dosyadan matrisin satır ve sütun sayıları okunsun. Sonra matris oluşturulsun ve dosyadan matris bilgileri okunsun. Dosyadan okunan tüm bilgiler okunmayla eşzamanlı olarak anlaşılır biçimde ekrana yazdırılsın. Okuma bitince dosya kapatılsın ve ekrana "A Matrisi Dosyadan Okundu" yazsın. En son programı sonlandırmak için bir tuşa basmak gereksin.

#include <iomanip>

#include <fstream>

#include <iostream>

using namespace std;

void main()

{

ifstream fr;

fr.open("file.txt");

if (fr.fail()) {

cout << "Dosya Açılamadı\n\n";

system("PAUSE");

return;

}

int m, n, M, N;

fr >> M >> N;

int \*\*a;

a = new int\*[M];

for (m = 0; m < M; m++)

a[m] = new int[N];

cout << setw(6) << M << setw(6) << N << endl;

for (m = 0; m < M; m++) {

for (n = 0; n < N; n++) {

fr >> a[m][n];

cout << setw(6) << a[m][n];

}

cout << endl;

}

fr.close();

cout << "A Matrisi Dosyadan Okundu\n\n";

system("PAUSE");

}

matrisleri boyutlu üç matris olsun. matris çarpımını hesaplayan altprogramı istenmektedir. Gerekli c programı kodları tamamlayınız.

const int n = 5;

float a[n][n], b[n][n], c[n][n];

void matris\_carp()

{int i, j, k;

for ( i = 0; i < n; i++ )

for (j = 0; j < n; j++)

{ c[i][j] = 0;

for ( k = 0; k < n; k++ )

c[i][j] += a[i][k] \* b[k][j];

}

}

n elemanlı tamsayı dizisinde en küçük elemanı bulan altprogramın bir kısmı aşağıda verilmiştir. Verilmeyen kısımları tamamlayınız.

int MinEleman( int dizi[], int n )

{int i, x = dizi[0];

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

if( x > dizi[i] ) x = dizi[i];

return x;

}

n elemanlı tamsayı dizisini klavyeden okuyan altprogramın bir kısmı aşağıda verilmiştir. Verilmeyen kısımları tamamlayınız.

void OkuDizi( int dizi[], int n )

{int i;

for( i = 0; i < n; i++ )

scanf( "%d", &dizi[i] );

fflush(stdin);

}

n elemanlı tamsayı dizisini ekranda gösteren altprogramın bir kısmı aşağıda verilmiştir. Verilmeyen kısımları tamamlayınız.

void YazDizi( int dizi[], int n )

{int i;

for( i = 0; i < n; i++ )

printf( " %d", dizi[i]);

printf("\n");

}

n elemanlı tamsayı dizisini klavyeden okuyan altprogramın bir kısmı aşağıda verilmiştir. Verilmeyen kısımları tamamlayınız.

void OkuDizi( int dizi[], int n )

{int i;

for( i = 0; i < n; i++ )

scanf( "%d", &dizi[i] );

fflush(stdin);

}

for ( i = 1; i <= n; i++ ) printf( "Say ---> %3d\n", i );

Yukarıda for döngüsü ile verilen kod parçasını while döngüsünü kullanarak yeniden yazınız. Ve do while döngüsünü kullanarak yeniden yazınız.

i = 1;

while ( i <= n ) printf( "Say ---> %3d\n", i++ );

i = 1;

do printf( "Say ---> %3d\n", i++ ); while ( i <= n );

-25 ile +25 arasında 20 tane rastgele tamsayı oluşturup ekranda gösteren bir program istenmektedir. Gerekli c programı kodları tamamlayınız.

void main()

{int i, j;

srand( time(NULL) );

for ( i = 1; i <= 20; i++ )

{ j = 25-rand()%51;

printf( "%5d.ci Rasgele Sayı = %3d\n", i, j );

}

printf( "\nÇıkmak için herhangi bir tuşa basınız" );

getch();

}

isimli dosyadaki bilgileri okuyup ekranda gösteren bir program istenmektedir. Gerekli c programı kodları tamamlayınız.

void main()

{FILE \*dosya;

dosya = fopen( "test.txt", "rb" );

if ( dosya == NULL )

{printf("Dosya açılamıyor\n");

getch();

return;

}

printf( "Dosyadaki bilgiler\n" );

printf( "------------------\n" );

char ch;

while ( true )

{ch = getc(dosya);

if ( feof(dosya) ) break;

printf( "%c", ch );

}

fclose(dosya);

getch();

}

Yığına veri ekleme, yığından veri çıkarma ve yığını ekranda gösterme olmak üzere üç tane altprogram istenmektedir. Gerekli c programı kodları tamamlayınız.

const int stack\_size = 5;

int stack[stack\_size];

int top = 0;

void Push\_Stack( int data )

{if ( top < stack\_size )

{stack[top] = data;

top++;

printf( "Push %d\n", data );

}

else printf( "Not Push, stack is full\n" );

}

int Pop\_Stack()

{if ( top > 0 )

{top--;

printf( "Pop %d\n", stack[top] );

return stack[top];

}

printf( "Not Pop, stack is empty\n" );

return 0;

}

void Print\_Stack()

{int i;

printf( "\nStack =" );

for ( i = 0; i < top; i++ )

printf( " %d", stack[i] );

printf( "\n" );

}

Kabarcık sıralama algoritmasını kullanarak n elemanlı bir tamsayı dizisini küçükten büyüğe sıralayan altprogramın bir kısmı aşağıda verilmiştir. Verilmeyen kısımları tamamlayınız.

void Swap( int &x, int &y ) // yer değiştir

{int z = x;

x = y;

y = z;

}

void BubbleSort( int dizi[], int n )

{int i, j;

do{

j = 0;

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

if ( dizi[i] > dizi[i+1] )

{Swap( dizi[i], dizi[i+1] );

j++;

}

} while ( j > 0 );

}

İkili arama algoritmasını kullanarak n elemanlı bir tamsayı dizisinde arama yapan altprogramın bir kısmı aşağıdadır. Kalan kısımları tamamlayınız.

int BinarySearch( int dizi[], int n, int aranan )

{int i = 0;

int bas = 0;

int son = n - 1;

while ( bas <= son )

{i = ( bas + son ) / 2;

if ( aranan == dizi[i] ) return (i+1);

else if ( aranan > dizi[i] ) bas = i + 1;

else son = i - 1;

}

return 0;

}

Seçmeli sıralama algoritmasını kullanarak n elemanlı bir tamsayı dizisini küçükten büyüğe sıralayan altprogramın bir kısmı aşağıda verilmiştir. Verilmeyen kısımları tamamlayınız.

void Swap( int &x, int &y ) // yer değiştir

{int z = x;

x = y;

y = z;

}

void SelectionSort( int dizi[], int n )

{int i, j, k;

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

{k = i;

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

if ( dizi[j] < dizi[k] ) k = j;

if ( k > i ) Swap( dizi[i], dizi[k] );

}

}

Aşağıdaki kod parçaları ne iş yapar açıklayınız.

clock\_t t1 = clock();

getch();

clock\_t t2 = clock();

printf( "%d ms", t2-t1 );

İstenilen milisaniye kadar bekletecek bekle isimli altprogramın bir kısmı aşağıda verilmiştir. Verilmeyen kısımları tamamlayınız.

void bekle( clock\_t sure )

{clock\_t t;

t = sure + clock();

while ( t > clock() );

}