**Это задание на курсовую по информатике.**

1)Нужно написать программу для решения диф. уравнений на языке С++. Предоставить к написанной программе схему алгоритма самой программы.

2)Построить графический объект (Букву В) и организовать его перемещения(масштабирование) по экрану монитора с помощью клавиш управления курсором. Это тоже на С++

3)Предоставить листинг программ.

4)Написать отчет о том как выполнялось задание (1,2).Описать ход работы.

**Ниже предоставлен пример курсовой работы.**

Міністерство освіти і науки, молоді та спорту України

Національний авіаційний університет

Факультет комп’ютерних систем

Кафедра комп’ютерних мультимедійних технологій

**КУРСОВА РОБОТА**

(пояснювальна записка)

з дисципліни

«Інформатика»

Тема:Розробка алгоритмічного та програмного забезпечення на мові С++

Варіант № 8

Виконав

Група:

Керівник: к.т.н., доцент

Київ 2012

Національний авіаційний університет

Факультет комп’ютерних систем

Кафедра комп’ютерних мультимедійних технологій

 Завдання

на виконання курсової роботи

студентки

 Тема курсової роботи: Розробка алгоритмічного та програмного забезпечення на мові С++ згідно з варіантами завдань.

1. Термін виконання: з р. до
2. Етапи роботи:

- вивчення та аналіз отриманого завдання;

- обґрунтування вибору програмного забезпечення;

- опис етапів процесу обробки завдання на ПК;

- виконання варіантів завдань:

2.1. Побудувати графічний об’єкт (парусник) і організувати його переміщення по екрану монітора за допомогою клавіш управління курсором.

2.2. Розробити програмне забезпечення в середовищі С++ для опису структури «текстовий редактор». Для кожної області перераховані параметри об’єкта. Серед параметрів обов’язковим є ключове поле, яке ідентифікує об’єкт. Організувати пошук в структурі по полям result і name.

- Оформлення пояснювальної записки до курсової роботи.

- Результат виконання.

Завдання видав 18.10.09 р. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Завдання прийняв до виконання\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Курсову роботу захищено з оцінкою\_\_\_\_\_\_\_\_\_\_\_\_\_

2

Лист

**Зміст**

Пояснювальна записка..…………………………………….………….....….....2

Зміст………………………………………………………………………………3

Реферат…………………………………………………………………………...4

Вступ……………………………………………………………………………...5

Порядок виконання завдання №1……………………………………….…..….6

Порядок виконання завдання №2 …………………………………….……..…8

2.1 Схема алгоритму…………………………………….…….………..….8

Висновки …………………………………………………………………….…..10

Список використаної літератури………………………………….………….…11

Додаток А…………………………………………………………………….…..12

Додаток Б……………………………………….………………………….…….16

**Реферат**

 Пояснювальна записка до курсової роботи «Розробка алгоритмічного й програмного забезпечення мовою високого рівня С++» містить сторінок, 2 додатки, 3 літературних джерела.

 Об’єкт дослідження – графічний об’єкт, база даних.

 Ціль роботи – розробка алгоритмічного та програмного забезпечення для опису структури, а також для реалізації пересування графічного об'єкта по екрану монітора за допомогою клавіш у графічному режимі.

 Метод дослідження – створення графічного об’єкта, організування пошуку в структурі по полям числового і символьного типу для об’єкту.

 Результати курсової роботи рекомендується використати для аналізу характеристик текстового редактору, який включає в себе фірму-виробник, кількість вікон, кількість шрифтів і ціну.

**Вступ**

C++ — мова програмування високого рівня з підтримкою декількох парадигм програмування: об'єктно-орієнтованої, узагальненої та процедурної. Розроблена Берном у 1979 році та названа «Сі з класами». Страуструп перейменував мову у C++ у 1983 р. Базується на мові С. Визначена стандартом ISO/IEC 14882:2003.

У 1990-х роках С++ стала однією з найуживаніших мов програмування загального призначення. Мову використовують для системного програмування, розробки програмного забезпечення, написання драйверів, потужних серверних та клієнтських програм, а також для розробки розважальних програм таких як відео ігри. С++ суттєво вплинула на інші, популярні сьогодні, мови програмування: С# та Java.

**Порядок виконання завдання №1**

Побудувати графічний об’єкт (парусник) і організувати його переміщення по екрану монітора за допомогою клавіш управління курсором.

Лістинг програми дивіться у додатку А (ст. 11).



**Результат програми:**



****

****

**Порядок виконання завдання №2**

Для кожної області перераховані параметри об’єкта. Серед параметрів обов’язковими є ключове поле, яке ідентифікує об’єкт. У кожного об’єкта є так само одне або кілька числових полів, за якими ймовірні звернення до об’єкта.

Організувати пошук в структурі по полям числового і символьного типу.

Текстовий редактор (назва, фірма – виробник, кількість вікон, кількість шрифтів, ціна).

Лістинг програми дивіться у додатку Б (ст. 11).

**Схема алгоритму**



**Результат програми:**



**Висновки**

У результаті виконання курсової роботи засобами С++ було створено дві різні програми.

Створені програми працюють. Перша програма створює малюнок з геометричних фігур, а друга організовує пошук з файла.

Під час виконання завдань було ознайомлено із основними можливостями програми та вивчено нові її функції.

В програмах використовуються функції вводу та виводу, оператори циклів. Були створені блок-схеми, програми перевірені за допомогою розрахунків.

В процесі виконання роботи використовувались такі програми, як Turbo C++ та Microsoft Word 2010.

Дана робота може використовуватись у навчальних цілях, з метою покращення знань у галузі програмування, а також створена програма може бути ефективно застосована в роботі з текстами для пошуку обраних слів.

**Список використаної літератури**

1. Полухін А.В. Методичні вказівки по курсовому проектуванню

/ М.С.Кулик, А.В.Полухін – К.: НАУ, 1999. – 36 с.

2. Полухін А.В. Положення про курсове проектування / М.С.Кулик,

А.В.Полухін – К.: НАУ, 2002. – 32 с.

3. ШибицькаН.Н*.* Алгоритмические языки и программное обеспечение //

Конспект лекций. - К.: КМУГА, 1998. - 160 с.

**Додаток А**

Лістинг програми створення графічного об’єкта

#include<graphics.h>

#include<conio.h>

#include<dos.h>

int offsetX = 0;

int DELTA = 4;

int MIN\_OFFSET\_X = -190;

int MAX\_OFFSET\_X = 190;

void drawBackground();

void drawBoat();

void cLine(int x1, int y1, int x2, int y2);

void cEllipse(int x1, int y1, int x2, int y2, int r1, int r2);

void cCircle(int x, int y, int r);

void cFloodFill(int x, int y, int color);

main()

{

int gd=DETECT,gm=VGA;

initgraph(&gd,&gm, "D\\TC\\BGI");

char c;

drawBackground();

drawBoat();

while(c!=27){ //Esc

while(!kbhit());

c = getch();

if(c==75 || c==77){

switch(c){

case 75: offsetX -= offsetX <= MIN\_OFFSET\_X ? 0 : DELTA; break; //left

case 77: offsetX += offsetX >= MAX\_OFFSET\_X ? 0 : DELTA; break; //right

}

drawBoat();

delay(50);

}

}

closegraph();

return 0;

}

void drawBackground(){

setbkcolor(11);

//more

line(0,360,650,360);

//colnze

circle(100,100,20);

line(100,130,100,180);

line(100,130,105,180);

line(100,180,105,180);

line(100,70,100,20);

line(100,70,95,20);

line(100,20,95,20);

line(130,100,170,100);

line(130,100,170,95);

line(170,100,170,95);

line(70,100,30,100);

line(70,100,30,105);

line(30,100,30,105);

line(115,70,130,30);

line(115,70,125,30);

line(130,30,125,30);

line(125,80,160,60);

line(125,80,150,60);

line(160,60,150,60);

line(115,130,130,170);

line(115,130,135,170);

line(130,170,135,170);

line(125,120,160,140);

line(125,120,150,140);

line(160,140,150,140);

line(85,130,70,170);

line(85,130,65,170);

line(70,170,65,170);

line(75,120,40,140);

line(75,120,50,140);

line(40,140,50,140);

line(85,70,70,30);

line(85,70,75,30);

line(70,30,75,30);

line(75,80,40,60);

line(75,80,50,60);

line(40,60,50,60);

//more

setfillstyle(SOLID\_FILL,1);

floodfill(340,380,WHITE);

//colnze

setfillstyle(SOLID\_FILL,14);

floodfill(100,100,WHITE);

floodfill(99,30,WHITE);

floodfill(101,150,WHITE);

floodfill(150,99,WHITE);

floodfill(50,101,WHITE);

floodfill(128,32,WHITE);

floodfill(152,62,WHITE);

floodfill(132,168,WHITE);

floodfill(152,138,WHITE);

floodfill(67,168,WHITE);

floodfill(43,139,WHITE);

floodfill(72,32,WHITE);

floodfill(48,62,WHITE);

}

void drawBoat(){

//remove old boat

setfillstyle(SOLID\_FILL,11);

bar(0, 180, 650, 359);

//korpus

cLine(245,360,400,360);

cLine(245,360,245,320);

cLine(245,320,340,320);

cLine(340,320,350,300);

cLine(400,360,430,300);

cLine(350,300,430,300);

cLine(245,320,251,310);

cLine(251,310,270,310);

cLine(430,300,430,295);

cLine(430,295,403,295);

//nadstroyka

cLine(270,320,270,270);

cLine(400,300,400,270);

cLine(270,270,400,270);

cLine(270,269,273,265);

cLine(273,265,290,265);

cLine(373,265,403,265);

cLine(403,265,403,300);

cLine(403,265,400,270);

//okno

cLine(280,280,320,280);

cLine(280,282,320,282);

cLine(280,302,320,302);

cLine(280,300,320,300);

cLine(280,302,280,280);

cLine(320,302,320,280);

//truba

cLine(290,270,290,230);

cLine(310,270,310,230);

cEllipse(300,230,310,225,10,6);

cLine(294,235,306,235);

//kapitanskiy mostic

cLine(310,250,370,250);

cLine(370,250,370,270);

cLine(310,248,372,248);

cLine(372,248,370,250);

cLine(372,248,372,270);

//machta

cLine(350,250,350,180);

cLine(348,250,348,180);

cLine(348,180,350,180);

//osnastka

cLine(350,180,430,295);

cLine(305,225,348,180);

cLine(272,265,290,244);

//illyuminatoru

cCircle(360,325,5);

cCircle(380,325,5);

cCircle(400,325,5);

//flagi

cLine(245,315,245,280);

cLine(245,280,225,285);

cLine(245,295,225,300);

cLine(225,285,225,300);

cLine(250,310,250,260);

cLine(250,260,230,266);

cLine(230,266,230,280);

cLine(230,280,250,275);

//korpus

setfillstyle(SOLID\_FILL,8);

cFloodFill(300,330,WHITE);

setfillstyle(SOLID\_FILL,7);

cFloodFill(420,298,WHITE);

cFloodFill(260,319,WHITE);

//nadstroyka

setfillstyle(SOLID\_FILL,6);

cFloodFill(340,300,WHITE);

cFloodFill(401,299,WHITE);

setfillstyle(SOLID\_FILL,7);

cFloodFill(399,269,WHITE);

cFloodFill(280,269,WHITE);

//okno

setfillstyle(SOLID\_FILL,14);

cFloodFill(300,290,WHITE);

setfillstyle(SOLID\_FILL,8);

cFloodFill(300,281,WHITE);

cFloodFill(300,301,WHITE);

//truba

setfillstyle(SOLID\_FILL,7);

cFloodFill(300,229,WHITE);

setfillstyle(SOLID\_FILL,8);

cFloodFill(300,240,WHITE);

//kapitanskiy mostic

setfillstyle(SOLID\_FILL,8);

cFloodFill(339,260,WHITE);

cFloodFill(371,269,WHITE);

setfillstyle(SOLID\_FILL,7);

cFloodFill(340,249,WHITE);

cFloodFill(360,249,WHITE);

//machta

setfillstyle(SOLID\_FILL,6);

cFloodFill(349,230,WHITE);

//illyuminatoru

setfillstyle(SOLID\_FILL,14);

cFloodFill(360,329,WHITE);

cFloodFill(381,329,WHITE);

cFloodFill(401,329,WHITE);

//flagi

setfillstyle(SOLID\_FILL,14);

cFloodFill(240,290,WHITE);

setfillstyle(SOLID\_FILL,1);

cFloodFill(240,270,WHITE);

}

void cLine(int x1, int y1, int x2, int y2){

line(x1 + offsetX, y1, x2 + offsetX, y2);

}

void cEllipse(int x1, int y, int stangle, int endangle, int r1, int r2){

ellipse(x1 + offsetX, y, stangle, endangle, r1, r2);

}

void cCircle(int x, int y, int r){

circle(x + offsetX, y, r);

}

void cFloodFill(int x, int y, int color){

floodfill(x + offsetX, y, color);

}

**Додаток Б**

Лістинг програми баз даних

#include<iostream.h>

#include<conio.h>

#include<stdlib.h>

#include<string.h>

#include<fstream.h>

struct name {

 char Name[20];

 char id[10];

};

struct description {

 char firma[20];

 int windows;

 int fonts;

 int value;

 char id;

 name\* FIN;

 description \*next;

};

description \*firms;

description \*first;

void catalog(char file[]);

void cataloginit();

void table();

void main() {

 clrscr();

 table();

 getch();

};

void table() {

 int k=0;

 while(k==0) {

 cataloginit();

 int tp;

 cout << "\n\n - find firms (1) - \n";

 cout << " - find firms by fonts (2) - \n";

 cout << " - find firms by windows (3) - \n";

 cout << " - find firms by value (4) - \n";

 cout << " - find firms by id (5) - \n";

 cout << " - exit (6) - \n";

 cout << " enter number-> ";

 cin >> tp;

 if (tp == 1) {

 char x[10];

 cout<<"\n enter description name - ";

 cin>>x;

 firms=first;

 int find = 0;

 while (firms!=NULL) {

 if (!strcmp(firms->FIN->Name,x)) {

 find=1;

 cout<<"\nNAME - "<<firms->FIN->Name;

 cout<<"\nfirma - "<<firms->firma;

 cout<<"\nwindows - "<<firms->windows;

 cout<<"\nfonts - "<<firms->fonts;

 cout<<"\nvalue - "<<firms->value;

 cout<<"\nid-"<<firms->FIN->id;

 };

 firms = firms ->next;

 };

 if (find==0) cout<<"\n not find..";

 }

 else {

 if (tp == 2) {

 int x;

 cout<<"\n enter description fonts- ";

 cin>>x;

 firms=first;

 int find = 0;

 while (firms!=NULL) {

 if (firms->fonts==x) {

 find=1;

 cout<<"\nName - "<<firms->FIN->Name;

 cout<<"\nfirma - "<<firms->firma;

 cout<<"\nwindows - "<<firms->windows;

 cout<<"\nfonts - "<<firms->fonts;

 cout<<"\nvalue - "<<firms->value;

 cout<<"\nid-"<<firms->FIN->id;

 };

 firms = firms->next;

 };

 if (find==0) cout<<"\n not find..";

 }

if (tp == 3) {

 int x;

 cout<<"\n enter description windows- ";

 cin>>x;

 firms=first;

 int find = 0;

 while (firms!=NULL) {

 if (firms->windows==x) {

 find=1;

 cout<<"\nName - "<<firms->FIN->Name;

 cout<<"\nfirma - "<<firms->firma;

 cout<<"\nwindows - "<<firms->windows;

 cout<<"\nfonts - "<<firms->fonts;

 cout<<"\nvalue - "<<firms->value;

 cout<<"\nid-"<<firms->FIN->id;

 };

 firms = firms->next;

 };

 if (find==0) cout<<"\n not find..";

 }

if (tp == 4) {

 int x;

 cout<<"\n enter description value- ";

 cin>>x;

 firms=first;

 int find = 0;

 while (firms!=NULL) {

 if (firms->value==x) {

 find=1;

 cout<<"\nName - "<<firms->FIN->Name;

 cout<<"\nfirma - "<<firms->firma;

 cout<<"\nwindows - "<<firms->windows;

 cout<<"\nfonts - "<<firms->fonts;

 cout<<"\nvalue - "<<firms->value;

 cout<<"\nid-"<<firms->FIN->id;

 };

 firms = firms->next;

 };

 if (find==0) cout<<"\n not find..";

 }

 if (tp == 5) {

 char x[10];

 cout<<"\n enter description id - ";

 cin>>x;

 firms=first;

 int find = 0;

 while (firms!=NULL) {

 if (!strcmp(firms->FIN->id,x)) {

 find=1;

 cout<<"\nNAME - "<<firms->FIN->Name;

 cout<<"\nfirma - "<<firms->firma;

 cout<<"\nwindows - "<<firms->windows;

 cout<<"\nfonts - "<<firms->fonts;

 cout<<"\nvalue - "<<firms->value;

 cout<<"\nid-"<<firms->FIN->id;

 };

 firms = firms ->next;

 };

 if (find==0) cout<<"\n not find..";

 }

 else if (tp == 6) break;

 };

 };

};

void catalog(char file[]) {

 ifstream firmsFile(file);

 while (firmsFile.good()) {

 firmsFile>>firms->FIN->Name;

firmsFile>>firms->firma;

 firmsFile>>firms->windows;

 firmsFile>>firms->fonts;

 firmsFile>>firms->value;

 firmsFile>>firms->FIN->id;

 firms->next = new (description);

 firms = firms->next;

 };

};

void cataloginit() {

 firms = new (description);

 first = firms;

 catalog("firms.txt");

};