O’ZBEKISTON RESPUBLIKASI
OLIY VA O’RTA MAXSUS TA’LIM VAZIRLIGI

ALISHER NAVOIY NOMIDAGI
SAMARQAND DAVLAT UNIVERSITETI

MEXANIKA-MATEMATIKA FAKULTETI

AXBOROTLASHTIRISH TEXNOLOGIYALARI
KAFEDRASI

RUSTAMOV RAMZIDDIN

ANTIVIRUS PROGRAMMALAR YARATISH MEXANIZMLARINI ISHLAB CHIQISH VA ULARNI QO’LLASH

5480100-Amaliy matematika va informatika ta’lim yo’nalishi bo’yicha bakalavr darajasini olish uchun

BITIRUV MALAKAVIY ISHI

Himoyaga ruxsat etildi:

Fakultet dekani: **prof. A. Soleev**

Kafedra mudiri: **dots. O. Djumanov**

Ilmiy rahbar: **dots. S. Qobilov**

Samarqand - 2012

Mundarija

KIRISH

Mavzuning dolzarbligi. Kompyuter xavfsizligi uning to'g'ri ishlashi, ma'lumotlarni to'liq va yaxshi saqlashi va programmalarni ishi buzulmasligi jarayonlaridan bog'liqdir. Shu jarayonlarga salbiy ta'sir qiladigan programmalarni programmaviy **viruslar** va ulardan himoya qilish uchun yaratilgan maxsus programmalarni **antiviruslar** deydilar. Bitiruv ishida ana shunday antivirus programmalar yaratish mexanizmlari va ularni qo'llash imkoniyatlari muhokama qilinadi.

Mavzuning dolzarbligi quyidagilar bilan asoslanadi:

antivirus programmalar yaratish va ishlatish kompyuter xavfsizligini ta'minlaydi;

kompyuter viruslari nafaqat ma'lumotlar va programmalarni balki, hisoblash tizmining apparat vositalarni ishdan chiqarish mumkin;

antivirus programmalar loyihalash amaliy informatikaning notrivial masalasi bo'lib, kompyuter arxitekturasi, tizimli programmaviy ta'minot funksiyalari va mashinaga - mo'ljallangan tillarni mukammal bilishni talab qiladi.

Hozirgi payitda tobora ko'p kompyuterlarning internetga ulanishi va ma'lumotlarning ozod ayiraboshlanishi mavzu dolzarbligini yanada ko'tardi.

Masalaning qo'yilishi. Bitiruv malakaviy ishida tizimli dasturlashning qiziqarli va muhim masalasi hisoblanadigan antivirus programmalar loyihalash mexanizmlarini ishlab chiqish va qo'llash yo'llarni ko'rsatish vazifasi qo'yilgan.

Bitiruv ishi maqsadi. Tizimli va amaliy programmalash usullardan foydalanib antivirus programmalarni loyihalash, taxlash va tadbiq etish bosqichlarini namoyish etish. Bu maqsadga erishish uchun quyidagi konkret ishlarni bajarish talab qilinadi.

- kompyuter viruslarining turlari, sinflash kodlari, bajariladigan funksiyalari va tarqalish muhitlarini tahlil qilish;

- antivirus yaratish mexanizmlarini o’rganish, himoyalanish texnalogiya sxemalarini tanlab olish va loyihalash instrumental vositalarini aniqlash;

- yaratilgan antivirus programmalarni ishlatilishini namoyish etish va qo’llashga doir ko’rsatmalar tayyorlash.

Tadqiqot metodlari. Ishni bajarish jarayonida tizimli va amaliy dasturlash texnologiyalari, bajariladigan fayllar strukturasini tahlil qilish usullari, ma’lumotlar strukturasini tasvirlash vositalar qo’llaniladi.

Ishning nazariy ahamiyati. Bitiruv ish dasturlash va programma ta’minotni loyihalash nuqtai nazaridan nazariy va texnalogik ahamiyatga ega. Unda antivirus programmalarni yaratishda viruslar turi, funksiyalari, ulardan himoyalanish usullari (programmani vaksinatsiya (emlash) qilish, exe va com fayllariga qulflar (belgilar) qo’yish, programma - filterlarni ishlatish) ketma-ket namoish etiladi.

Ishning amaliy ahamiyati. Bitiruv ishi amaliy va o’quv – uslubiy ahamiyatga ham ega. Viruslar va antiviruslarni sinflash prinsiplarini, antivirus programmalarni yaratish bosqichlarini, algoritmik yechimlarini o’quv yurtlari yuqori bosqich talabalaridan dasturlovchilar tayyorlashda maxsus kurs materiallari va uslubiy ko’rsatmalar sifatida ishlatish mumkin. Yaratilgan programma ta’minoti bajariladigan fayllarni himoyalash uchun qo’llanilishi maqsadga muvofiqdir.

Ish strukturasi. Bitruv malakaviy ishi to’rtta bo’lim, xulosa, adabiyotlar ro’yxati va ilovalardan iborat.

Birinchi bo’limda masalaning qo’yilishi va uning tahlili keltirilgan. Ikkinchi bo’limida kompyuter viruslarini sinflanishi, bajariladigan funksiyalari ulardan himoyalanishi ba’zi bir usullari muhokama qilinadi. Antivirus programmalarni yaratish sodda mexanizmlari, ularni programma shaklida loyihalash bosqichlari, instrumental vositalarni tanlash talablari uchinchi bo’limda to’liq yoritilgan.

To’rtinchi bo’limda esa, exe va com fayllarni himoyalash uchun qo’llaniladigan programmalarni ishlab chiqarilishi jarayoni batafsil berilgan.

Xulosada asosiy natijalar, adabiyotlar ro’yxatida ishlatilgan materiallar va Internet - resurslar, ilovalarda esa, programma matinlari berilgan.

1. MASALANING QO’YILISHI VA TAHLILI

Kompyuter tizimlarning rivojlanishi, turli sohalarda ishlatilishi va murakkablashish ular muhitida saqlanadigan ma’lumotlar hajmi va muhimligini himoya qilish masalalari dolzarbligi oshib bormoqda. Yangi programmalar, tizimlar va sohalar shaxsiy kompyuterlar uchun paydo bo’lmoqda. Ana shunday sohalardan biri ,, kompyuter bu rusolagiyasi” deb nomlanadi. Uning kelib chiqish asosiy sabablardan biri kompyuter viruslarni, ya’ni programmalarini yangi turi paydo bo’lishidir. Bunday **virus - programmalarga** qarshi kurashadigan **antivirus –programmalarni** hozirgi paytda bir nechta yirik firmalar ishlab chiqaradi.

Viruslar bajariladigan fayllarni (.com,.exe,.ovl), yuklash (boot) sektirlarni, drayverlarni, paketli (.bat) fayllarni va boshlang’ich programmalarni ishdan chiqarilishi mumkin[1]. Viruslarning turlari va soni tobora oshib bormoqda, ularni himoyalash usul va vositalari ham murakkablashmoqda. Viruslarni o’rganish, tahlil qilish, ularga qarshi antivirus programmalar yaratish dolzarb masala hisoblanadi.

Shularni hisobga olib, malakaviy bitiruv ishida antivirus programmalash yaratish mexanizimlarni ishlab chiqish va ularni qo’llash masalasi qo’yilgan. Masalaning bunday qo’yilishi konkret ishlarni bajarishni talab qiladi.

- kompyuter viruslari ularning xususiyatlari, sinflanishi va bajariladigan funksiyalarni tahlil qilish;
- antiviruslarni loyihalash va programmalash;
- yaratiladigan programma ta’minotini ishlash prinsiplarni namoyish etish.

Keltirilgan va ular bilan bog’liq bo’lgan qismiy masalalarni hal qilish amaliy informatikaning notirivial muammolari hisoblanadi.

Birinchidan, paydo bo’lgan yangi virus turini aniqlash murakkabdir. Ikkinchidan, agar virus konbinatsiyalashgan, ya’ni ham faylni va ham

disklarning yuklash sektorni ishdan chiqaradigan bo’lsa, unga qarshi antivirus strukturasi ham murakkab bo’ladi.

Uchinchidan, ba’zi bir viruslar, masalan, apparatura va katta ROM – BIOS programmasiga ta’sir qiladigan viruslar ishi natijasida operatsion tizim yuklanmaydi va ish to’xtab qoladi. To’rtinchidan, antivirus programmalarni loyihalash dasturlovchilardan kompyuter arxitekturasini, mashina va mashinaga mo’ljallangan tillarning va zamonaviy texnologiyalarni mukammal bilishni talab qiladi. Beshinchidan, viruslarning turi va soni har kun oshib bormoqda. Bu esa antivirus programmalarni ishlab chiqarish texnologiyalarni yanada mukammallashtirish davr talabi ekanini ko’rsatadi.

Ta’kidlash joizkim informatikaning kompyuter viruslari va antiviruslari mavzusini bitta bakalavrlik bitiruv ishida to’liq va har tomonlama muhokima qilish mumkin emas. Shuning uchun bu malakaviy bitiruv ishida qiziqarli va katta mavzuning bir muhim qismiy masalasi – mavjud bajariladigan (.exe) fayllarni himoya qilishning mumkin bo’lgan bitta usuli va uni programmalash yo’llari berilgan.

Bitiruv ishi mavzusining yana bir muhim tamoni shundan iborat kim Internet tarmog’ining juda tez rivojlanishi va millionlab kompyuterlarning bir-biri bilan ulanishi kompyuter viruslari muommolari yechishni tashkil etuvchi mutaxassislarni tayyorlashni jadallashtirishni ko’rsatadi. Hozirgi paytda bakalavriatura ta’lim yo’nalishida faqat kompyuter va axborot xafsisligi fanlarda keltirilgan muommolar muhokama qilinadi. Maxsus shu soha bo’yicha ta’lim yo’nalishlari va mutahassislari mavjud emas. Mavzu adabiyotlarda ham kam yoritilgan va informatikaning qo’shimcha boblari hisoblanadi. Shuning uchun keyinchalik kompyuter virusologiyasi sohasida ishlaydigan mutahasislarni tayorlashda tizimli dasturlash, kompyuter arxitekturasi, aperatsion sistemalar tarkibi va loyihalanishi, mashinaga – mo’ljallangan programmalash tillari, tarmoq texnolagiyalari, kodlashtirish va kobinotorika fanlarga chuqurroq e’tibor berish talab qilinadi.

Ishning natijalari, ya’ni ko’rsatmalar texnologik yechimlar va xulosalarni dasturlavchilarni tayyorlashda o’quv qo’llanmalar va materiallar sifatida va raqobatbardosh programma mahsulotlarni ishlab chiqarishda ishlatish maqsadga muvofiqdir.

2. KOMPYUTER VIRUSLARNING SINFLANISHI

Kompyuter viruslari mavzusi va ulardan himoyalanish muommolari muhimligini hamma tan oladi. Shuning uchun soha mutahasislari orasida ma’lumotlarni ayra boshlash va tahlil qilish asosiy vazifalardan biri bo’lib qoldi. Kompyuter viruslarni sinflash, ularni strukturasi va funksiyalarni o’rganish va ular bilan qarshi ko’rashish usul va mehanezimlarini ishlab chiqish informatika va programmalash sohasining dolzarb masalasidir. Bu masalani hal qilishda bir qancha muommalar mavjud. Masalan, virus turli xarakteristikalarda (ishlab chiqarish joyi,virus tanasidagi ASCII - satr, virus hosil qiladigan har xil effiklar), murakkab nomlanishga va shtammlarga (turlar, ko’rinishlar) hos bo’lishi mumkin.Bular va boshqa hususiyatlar viruslarni sinflashga katta etibor berishga majbur qiladi.Hozirgacha viruslarni sinflashga turlicha yondoshadilar. Masalan, sinflash sxemasi uchta quyidagi elimetlarni o’z ichiga olishi mumkin.[1,2,3]:

-Virus kodi(tranzistorlarni sinflash sxemasiga o’xshaydi);

-Virus deskriptori(asosiy xususiyatlari ruyxati);

-Virus signaturasi(zararlangan programmada izlash uchun maxsus satr).

Mazko’r sxema bo’yicha virusga **kod** beriladi.Kod **prefikisdan, sonli xarakteristikadan** va yordamchi (fakultatif) harfli **suffiksdan** iborat. Masalan, RSE-1813C nomga ega bo’lsin.Bu erda RSE-prefiks,1813-xaraktiristika va C-suffiks hisoblanadi.Sinflashning yana bir asosiy vazifasi zararlanmagan kompyutarda virus xususiyatlarni tahlil qilishidir.

Kompyuter viruslarni **tarqalish muhiti** (sreda razminojeniya)

bo’yicha ularni beshta asosiy guruhlarga bo’lish mumkin[4]:

Fayl viruslari.Ular com.exe.ovl fayllarni zaharlaydi.Prefikslari C,E,O.

Boot-viruslar. Boshlangich yuklash sektorni (Boot vektor) yoki MBR (master Boot recod) sohasini zaharlaydi.Prefikslari B,D,M.

Draywer viruslari qurilmalarini yoki konfig.sys fayliga o’rnashib zaharlaydi.Prefiks S.

Paketli viruslar. Paketli(.bat)fayllarni zaharlaydigan viruslar.Prefiks J.

Kompliyatsiyalanadigan viruslar. Kompliyatsiyalanadigan (tarjima qilinishga tayorlangan) programmalar (masalam, .pas, .asm,.cpp) matniga kirib zaharlaydigan viruslar. Prefiks T.

Tarmoq texnologiyalari va internetning tez rivojlanish natijasida bu muhitda paydo bo'lgan viruslar ham alohida sinf sifatida ajratiladi va o'rganib kelinmoqda.

Yuqoridalarni hisobga olib kompyuter viruslarni yana bir to'liqroq tarifini berish mumkin. **Kompyuter virusi informatsion muhitda o'z-o'zidan tarqaluvchi programmaviy koddir.**

Viruslar aniq vaqt o'tgandan keyin, yoki surinkali ma'lum vaqt va sharoit paydo bo'lganda ham ishni boshlash mumkin.

Viruslarni viruslarni tahlil qilish,sinflash va xaraktiristikalarni o'rganish ularni quyidagi ishlarni bajarish bilan yaqqol ko'rinadi dab hulosa qilish mumkin.

Ekranda begona yozuvlar paydo bo'lish;

Ekrandagi simvollarni ,, to'kilishi ";

Kompyuter qayta yuklanishi (ulanishi)(prizagruzka qilinishi);

Kompyuter ishini sekinlanishi;

Har xil musiqa va segnallarni (tavushlarni chiqarilishi);

Fayl va katolaglarni yo'qolishi qattiq va yumshoq disklarning siktorlarning ishdan chiqishi (buzilishi,o'chirilishi) va hakozalar.

Endi viruslarning asosiy turlari va hususiyatlarni ko'rib chiqamiz.

a) **Rezident va norezident viruslar.** Rezidant programma ishini to'gatgandan keyin boshqaruvni operatsion sestemaga (OS) beradi, ammo xotirada qoladi. Agar mahsus klavishlar bosilsa yoki sharoitlar bajarilsa rezident programma yana aktiv bo'lishi mumkin.Rezident virus ham operativ xotirada qolishi bilan haraktirlanadi. Shuning uchun rezidint virus operativ xotirani egallaydi. Xotirani taqsimlashning tekshirib virus, ya'ni rezident virus egallagan

xotira hajmini aniqlash mumkin. Shuning uchun yuklanuvchi viruslarning qarib hammasi va fayl viruslarning ko’pi rezident shaklida tashkil qilingan virus-programmalaridir. Bu tipdagi viruslardan himoyalanish uchun OC ni yangi (toza) sistemaviy diskdan yuklash maqsadga mofiqdir.

b) Disk yuklovchi sektoridagi viruslar. Malumki kompyuterni tok

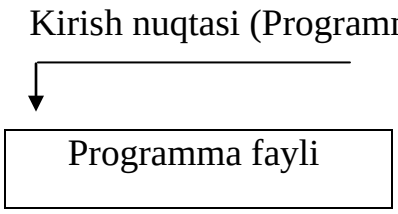
bilan ulagandan keyin birinchi bo’lib POST(Pover On Self Test) programmasi ish boshlaydi. Bu programma doimiy xotirada (PZU-ROM) yozilgan bo’lib. Qismiy sistemalar ishi va kompyuter konfigurasiyasini tekshiradi.

POST ishi davomida A:dickavodida disketa bo’lsa OC o’shandan yuklanadi, aks holda OS qattiq diskdan yuklanadi. Yuklochi viruslar dicketa yoki disk yuklovchi sektoridagi ma’lumotni o’zgartiradi. Yuklovchi virus operativ xotiraga kirib olganidan keyin boshqa programmalar va dirayverlarni ishini tekshirib turishi mumkin.

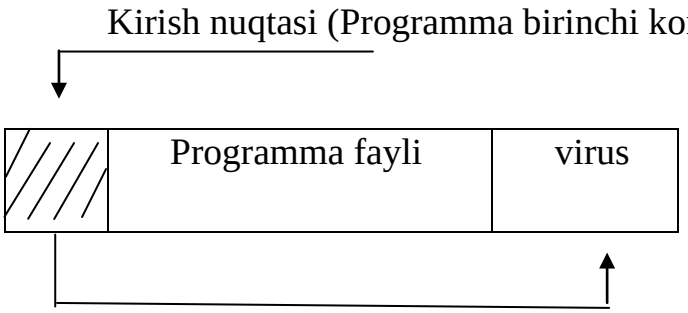
v) **Fayllarga joylashadigan (o’rnashadigan) viruslar**. Bu turdagi viruslar juda ko’pdir. Ular avvalo bajariladigan kodlarni saqlovchi fayllarga joylashadi. Bunday fayllarni asosiy qismini .com va exe fayllari tashkil qiladi. Bundan tashqari, overlay fayllari (ovl,.ovr-kengaymali fayllar) va dirayverlar (.sys) ham zararlanishi mumkin. .COM va EXE fayllari o’z formatlari bilan farq qiladi

COM faylli prosessor kamondalari va ma’lumotlardan iborat, EXE fayllari esa murakabroq strikturaga ega. EXE fayilida prasesor kamandalaridan tashqari maxsus **sarlovha** (заголовок,header) mavjud. Sarlovha fayil bosh qismida joylashgan. Agar .exe fayilini matn muxariri (riktori) yordamida kursor sarlovhaning birinchi maydoni ,,MZ” yoki ,,ZM” simvollarini saqlaydi. Sarlovxaning **taxlash jadvali** (таблица настройка) joylashadi viruslar shu maydonlarga zarar yetkazishi mumkin.

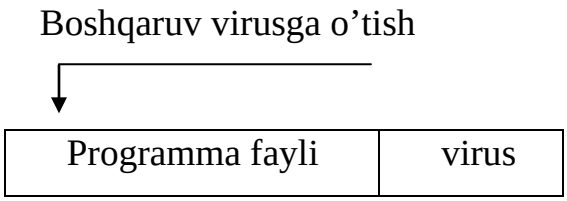
Komandali (.bat) fayllar OCning komandalar to’plami, tayyor programmalar va izohlardan iborat; Komandalar fayllining zararlanishi quyidagi bosqichlardan iborat



1-Bosqich. Zararlanishdan olingan programma fayl virusi bilan zararlanishi natijasida virus kodi fayl ichiga o’rnashadi.



2-Bosqich. Virus programma faylining birinchi komandasini o’zgartiradi.Fayl ish boshlagandan keyin boshqaruv virus kodiga o’tadi.



3-Bosqich. Boshqaruvning virusga o’tishi fayllarni viruslar bilan zaharlanishi quyidagi holatlar bilan namoyon bo’ladi.

h1) Programma taminoti va ma’lumotlar fayllarni bo’zilishi yoki noto’g’ri ishlashi.

h2) Fayl o’chirilishi faylga begona (musir)ma’lumotlarni kiritilishi.

h3) Disk farmatlanishi va ma’lumotlarning yo’qolishi.

h4) Disk sektorini kodlavchi (shifrlovchi) masalan, OneHalf virusi ularning ishidan keyin ma’lumotlarni o’qish bo’ziladi. Bu viruslar **стэлс-viruslar** ham deyiladi va ularni yo’qotish murakkablashadi.

h5) Apparatura ishini zararlantiradigan viruslar. Ular masalan, ROM-BIOS xotiradagi ma’lumotlarga zarar etkazadi va kompyuter yuklanmasligi mumkin.

h6) Odam ruhiyatiga (psihasiga) tasir qilqdigan funksiyalarni bajaruvchi viruslar. Masalan, kinokadr 25chi kadrdan o’zgarsa ish holatda odamga tasir qiladigan kadr yuborish mumkin. Hozirgi kompyuterlar bir sekuntdda 40gacha kadrni o’zgartirish mumkin.

h7) ,,mantiqiy bumba” tipidagi viruslar bajaradigan ishlar... ular bilan bir vaqtgacha ko’rinmaydi ma’lum bir vaqt kelganda o’z funksiyalarni bajarishga kirishadi. Masalan, Michelangelo 6 marta Jerusalim (,, qora juma “) juma kuni 13 sanaga to’g’ri kelsa fayllarni o’chiradi.

h8) Hujjat fayllardagi (.dos) viruslar. Bu viruslar hujjat matni bilan saqlanadigan **makrokamandalarini** o’zgartiradi va hujjatni qayta ishlash murakkablashadi.

Biz bu bo’limda konpyuter viruslarni sinflanishi masalalarni muhokama qildik. Sinflash sxemalari, virus kodi, viruslarni tarqalish muhiti hususiyatlari bilan tanishdik. Viruslarning asosiy turidan rezident va norezident viruslar bo’lib disk yuklovchi siktorga o’rnashadigan fayllarga joylashadigan viruslarni funksiyalari ko’rib chiqildi.

3. ANTIVIRUS PROGRAMMALAR YARATISH MEXANIZMLARI

Kompyuter viruslardan himoyalanish jarayonida ularning tarqalishi va ish bajarishi mexanizmlarni o’rganish asosiy vazifa hisoblanadi. Bu vazifani hal etish, viruslarga qarshi ko’rashishning bosh omili desak xato qilamiz.ixtiyoriy virus mavjudlik ya’ni ,, yashash” sklida uchta bosqichni ajratish mumkin.[5].

Birinchi bosqichda virus **noaktiv** holatda bo’ladi u bajaradigan fayl tanasi yoki diskni yuklash (загрузочный) siktorida kritilgandir. Anashu holatda virus programmalar yoki diskitalar (flish-xotiralar)yordamida bir SHK dan boshqasiga o’tadi.Noaktif holatda virus hali o’z ,,ishini “ boshlamaydi.Virus o’z ishinni boshlashi uchun bajaradigan faylni ishga yuklash yoki zaharlangan diskitadan yuklanishi kerak bo’ladi. Bular bajarilsa virus aktivlanadi. U xotirada rezident programma hosil qilishi, undan nushalar paydo bo’lishi, to’rli ,, bo’zg’unchilik” amallarni boshlanishi mumkin.

Agar virus rezident programmani hosil qilgan bo’lsa, uning aktivlanishi turli yo’llardan iborat bo’ladi.Bu yo’llar virus ishlab chiqqan muallif-dasturlavchining tanlab olingan usullardan bog’liqdir. Faraz qilaylik, MS_DOS muhitida viruslarning juda ko’p turlari ishlab chiqarilgan. Ularning asosiy qismi $21 uzilishga tasir qiladi chunki $21 nomerli o’zilish operatsion tizimda ixtiyoriy operatsiyalarga, ya’ni MS_DOS bilan bajariladigan amallarga murojat qilishga asosiy hisoblanadi.

Ikkinchi bisqichda ya’ni virus boshqaruvini o’z quliga olganda (resident programma aktivlashganda ham) virus ,,ko’payishga “ holatga o’tadi. U kerakli bajariladigan programma tarkibiga (tanasiga) o’z kodini kritadi. Foydalanuvchilar virusni sezmasligi uchun virus har bir aktivlashganda faqat bitta faylga ,,o’rnashadi” .Qadamma-qadam virus tabora ko’p fayllarga o’rnashib o’z ishini davom ettiradi.

Uchinchi bosqichda, ya’ni yetarlicha ko’p fayllarni virus tamondan zaharlangandan keyin virus ishining tashqi ko’rinishlari (simptomxari) namoyon bo’ladi. Anashu holda kompyuter ishida o’zgarishlar seziladi. Masalan, qandaydir musiqa progmentining eshitilishi, simvallarning manitor ekranda ,,to’qilishi” va hakozalar. Bunday uzgarishlar fayil tuzilishining buzulishidan darak beradi.

Viruslarni lakalizatsiya qilish (aniqlash) va o’chirish uchun turli antivirus programalari (Aedstest, nod 32, kasperiski antiviruslari, Norton antiviruslari) qo’llash maqsadga muvofiq. Agar antivirus programmalar bo’lmasa, u holda kerakli qattiq diskdan muhim fayllarni (hujjatlarni) nusxasini olish kerak. Ammo bajariladigan (.exe va .com) fayilaridan nusxa olish kerak emas. Bu ishlar bajarilgandan keyin qattiq disk qayta formatlanadi, operatsion tizim modelari urnatiladi va saqlab qolingan fayllar disketadan qattiq diska yoziladi.

Programalarni ,,vaksinatsiya” qilish (emlash).

Virus programalardan himoyalanish va yuqoridagi holatlarga tushmasligi uchun asosiy va muhim programalardan doimo disketalardan nusxa olib qo’yish maqsadga muvofiqdir. Nusxalarni arxivlab qo’yish katta ahamiyatga ega.

Shular bilan bir qatorda programmalarni ,,vaksinatsiya” qilish (emlash) usulari mavjud. Uning asosiy g’oyasi quyidagicha sharqlanadi [6]. Bajariladigan fayllarni vaksinatsiya qilib turish uchun ularga **o’z-o’zini tekshirib turish,** ya’ni **diognostika qilish** xususiyatlarini kiritadilar. Agar zararlanish fakiti aniqlansa programa o’zining **boshlang’ich holatini tiklashga** o’tish kerak. Surinkali tekshirishlar (vaksinatsiyalar) virusga ,,immunitet” paydo qilish xususiyatga ega bo’lgan programalarni yaratishga imkoniyat beradi.

Bu usulning ko’psonli antivirus programmalar yaratish usullardan yana bir qulayligi shundan iboratki, foydalanuvchi o’z programmasida qandaydir mahsus ma’lumotni kiritadi. Programma ixtiyoriy virus bilan zararlanganda uni ishga tushirish paytida juda tez namayon bo’ladi. **Zararlanmagan** fayllar haqida qaysi informatsiyani saqlash va keyin uni tekshirish kerak? degan qiziqarli savol payda

bo’ladi. Bu savolga javob berish uchun biz bajariladigan (.exe va .com) fayllarni formati va tuzilishini tahlil qilishimiz kerak.

Faraz qilaylik, bizga ma’lum bo’lgan Pascal oilasiga qarashli programmalash tizmi berilgan bo’sin. Programma, ya’ni .pas kenkaymali fayl tarjima qilingandan keyin .exe faylga o’tkaziladi.

Bajariladigan fayllar sarlovhasi. Bunday fayllarning bosh qismida **sarlovha** joylashadi. Sarlovha diskdagi programmani ishga yaroqli (tayyor) programmaga o’tkazish uchun kerak bo’lgan tuliq ma’lumot (infarmatsiya) saqlanadi. Sarlovhaning birinchi 28 bayt quyidagi straturaga ega.

HeadExeType=**record**

Sign:**word**; {EXE-fayl belgisi}

PortPag:**word**;{fayl oxiriga to’liqmas sector}

PageCnt:ward;{Sektorlar soni}

ReloCnt:word;{Kritish jadvalidagi elimentsoni}

HdrSize:word;{Sarlovha uzunligi}

MinMem:word;{Kucha minimal o’lchami}

MaxMem:word;{Kucha maksimal o’lchami}

Reloss:word;”{Steg segment boshlangich qiymati}

Exesp:word;{Stek ko’rsatgich qiymati}

ChkSum:word;{ Fayl so’zlari yig’indisi }

ExeIP:word;{Programma ish boshlash nuqtasi}

Relocs:word;{Kod segminti boshlang’ich qiymati}

TablOff:wosr;{Birinchi elimetn ko’rsatgichi}

Overlay:word;{Overlay nomeri

end;

Sarlavhaning qolgan elimentlari **ko’chirish jadvalini** saqlaydi. Jadval TablOff baytidan boshlanib ReloCnt to’rtbayti elimentlarini o’z ichiga oladi. Uning umumiy ko’rinishi quidagichadir:

type

ReloTablIrem=**record**

ItemSeg:word;{Siljish adresi segmenti}

ItemOfs:word:{Siljish adrisi ko’chishi}

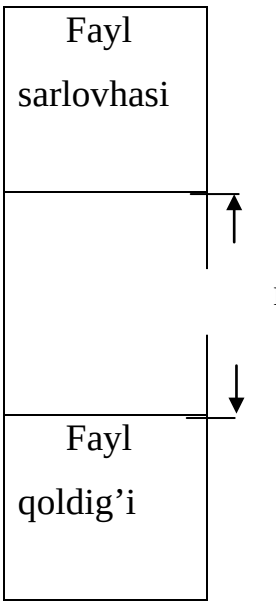
end;

Exe-fayli belgisi Sign maydonida ,,MZ”(kodi$5A4D)simvoli ko’rinishida saqlanadi.Bu belgi yordamida ixtiyoriy EXE fayl boshlanishi kerak. HdrSize sarlovhani to’liq uzunligini saqlaydi. Uzunlik paragraflar, ya’ni har biri 16 bayt uzunlika ega bo’lgan xotra sohasi yordamida ulchanadi. PartPage va Pagecnt maydonlari EXE-fayilining xotraga yuklanadigan umumiy uzunligini quyidagi formula bilan aniqlaydi.

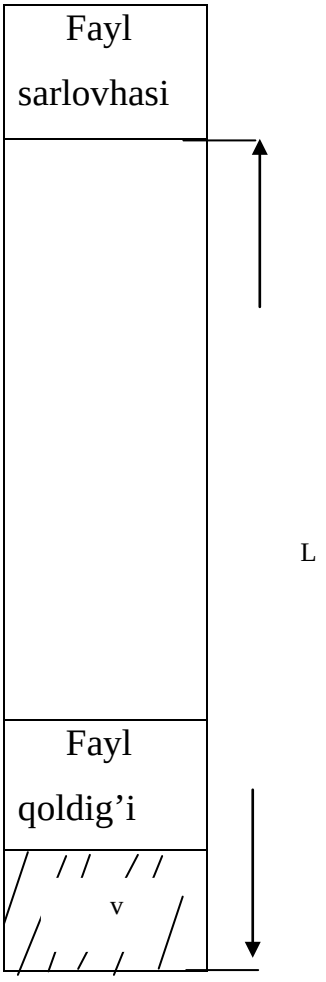
α=(PageCnt-1)*512+PartPagi*-Hdr+Site*16.

Exe-viruslarni qarib hammasi o’z programmasini (kodini) **faylning** (.exe-faylning) **ohirga** joylashtiradi.Xotiraga kirish va unga ya’ni virusga bashqaruvni berish uchun PartPagi, PagiCnt,ReloCS,ExeIP maydanlarni o’zgartiradi. Virusning bunday **o’rnashishida** faylning xotiraga yuklanadigan umumiy uzinligi ExeSize=FileSize+VirusSize formula yordamda hisoblanadi. Bu erda FileSize EXE faylning umumiy uzunligi, virus Size esa virus programmasi uzunligidir. ExeSize uzunligi juda katta bo’lib ketsa u xotiraga yuklanmaydi. Ana shu holatda virus o’zini namayon qiladi va zaharlangan programma ishini bashlolmaydi.

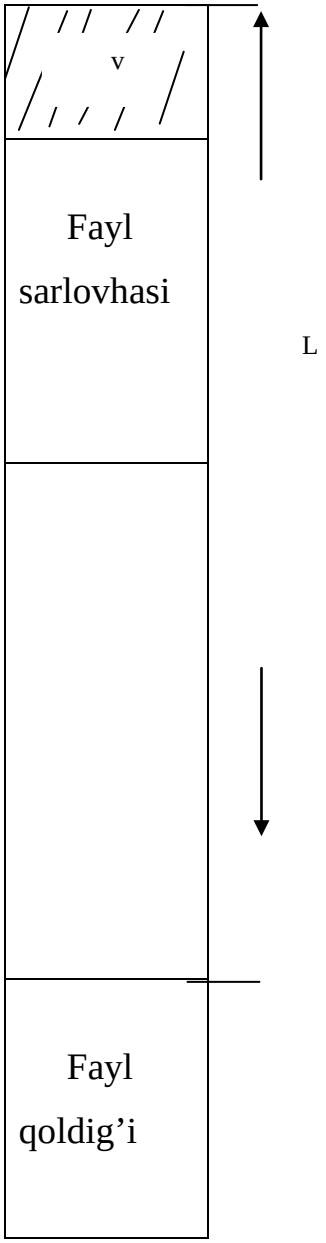
Umuman olganda, viruslar exe-fayllarga quyidagi sxemada ko’rsatilgan usullar yordamda o’rnashadilar.


Fayl
sarlovhasi
Fayl
qoldig’i

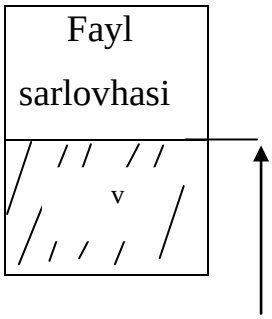
a)

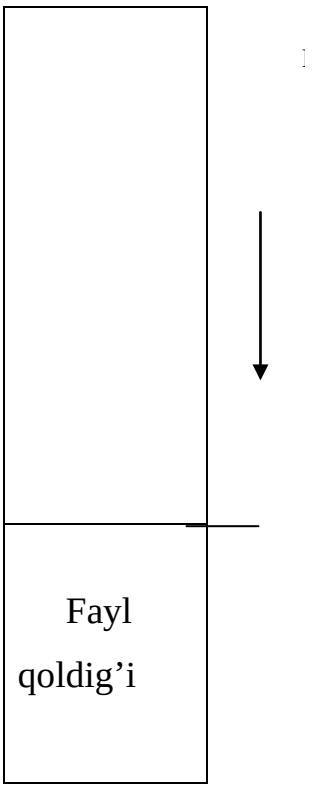

Fayl
sarlovhasi
L
Fayl
qoldig’i
v

b)



v)





g)

Virusning fayl tarkibiga o’rnashish variantlari.

a) zaharlanmagan fayl;

b) virus fayl ohrida;

v) virus fayl boshida;

g) virus fayl o’rtasida.

α-faylning xotiraga yuklanadigan qismi.

Agar biz (g) sxemani tahlil qilsak virus fayl sarlovhasidan keyin joylashadi. Fayl uzunligi virus kodi uzunligicha kengayadi. Bu usul juda kam qullanadi, chunki bunday viruslarni liyihalash juda murakkab hisoblanadi. Virus o’z ishini tashkil qilish uchun ko’p vaqt talab qilinadi va foydalanuvchi tamondan uni aniqlanishi (topilishi)osanlashadi.

Umumiy tahlil shuni ko’rsatadiki, mavjud yoki loyihalanadigan exe-viruslarning qarib hammasi EXE – fayllarni ularning ohiriga o’rnashiob zaharlaydi va fayl sarlovhasini o’zgartiradi. Ana shu holat asasiy hulosaga ya’ni

programmani virus tamondan zaharlanishini aniqlash va virusni yuqatish uchun biz quidagi ishlarni bajarish kerak ekan.

fayl sarlovhasi va uning etalan (asil) uzunligini qayirdadir saqlash.

Surinkali (davriy tarzda) fayl joriy uzunligi va sarlovhasini etalan ko’rsatgichlari bilan solishtirib turish.

Shularni hisobga olib yangi loyihalanadigan programmalarni himoya qilish uchun quyidagicha ish ko’rish mimkin.

Zaharlanmagan ya’ni virus tasir qilmagan programma haqidagi asosiy ma’lumotni (uni kalit yoki kalitli ma’lumot diymiz) alohida faylda saqlash mumkin. Ammo bu faylni qayta ishlash yoki nushalash paytida yuqotishimiz yoki buzishimiz mumkin. Agar biz kalit haqidagi ma’lumotni faylni o’zida, ya’ni himoyalanishi kerak bo’lgan faylni o’zida saqlolsak ishimiz ancha yingil bo’ladi. Buni tashkil qilish uchun yullardan biri programmada **tiplashtirilgan konstantani** aniqlab kalit ma’lumotni unda saqlashning tashkillashtirish maqsatga muvofiqdir. Ammo tiplashtirilgan konstanta qiymatni izlash katta EXE fayllarda murakkablashadi. Bu murakkablikni soddalashtirish uchun fayl boshidan kalitgacha adrislar siljishini aniqlash usulini qullaydilar. Har bir programma PSP-programma sigmenti priviksi bilan bog’liqdir. Unda operatsion tizim va programma uchun yordamchi ma’lumot saqlanadi. Operatsion tizimning $62 uzilishi PSP sigmenti qiymatini VX regestiriga qaytaradi. Shundan foydalanib ma’lumotlarni saqlash sohasi adrislarnni aniqlash mumkin buladi. Endi faqat bizga kerakli bo’lgan kanstantani aniqlash masalasini oladi.

Exe fayllarda, masalan,tarjima qilingan Paskal - programmada konstantalar e’lon qilingan tartibda saqlanadilar. Konstanta **o’lchami** va e’lon qilinganligi **tartibini** aniqlab bizga keraklisini topish mumkin. Ikkinchi tamondan bizga adrisni aniqlash operatsiyasi - C ham yordam qiladi. Bu operatsiyani qullash natijasida 4 baytli aders aniqlanadi va u bizga karakli kalit adris va ma’lumotlarni saqlash sohaga nisbattan joylashishi ko’rsatiladi.

Shunday qilib, biz bu bo’limda antivirus programa yaratish mexanizmlarini muxakama qildik. Buning uchun ketme-ket quyidagi masalalarni holati va ularni yechish bosqichlarini taxlilini berdik.

Viruslarni ,,yashash” va ishlash siklini ajratish;

Rezident viruslarini aktivlashtirish yo’lari;

Viruslarni lakalizatsiya qilish va ularni o’chirish;

Programalarni ,,emlash” yo’larini belgilash;

Bajariladigan (.exe-fayilar) strikturasini tekshirish;

Yangi va mavjud fayilarni zaxarlanishini oldini olish mexanizmlarini muxakama qilish.

4. ANTIVIRUS PROGRAMMALARNING QO’LLANISHI

Biz bu bo’limda ya’ni tashkil qilinadigan EXE - fayllarni himoya qilish uchun antivirus yaratish ketma-ketligini muhokama qilamiz. Asosiy g’oya EXE-faylini yaratish jarayonida uning tarkibiga maxsus **kalitni** kiritish va keyinchalik bu faylni har bir qo’llanilishida kalitni tekshirish bilan bog’liqdir.

Antivirus programma modul shaklida tashkil qilinadi. EXE-faylni zararlanishini tekshirish kaliti sifatida **head** tiplashtirilgan konstanta ishlatiladi. Uning umumiy ko’rinishi quyidagi struktura bilan beriladi.

```
type
HType=record
        HE: HeadExeType; {fayl sarlovhasi}
        HL: LongInt; {fayl etalon uzunligi}
        HF: Boolean; {Kalitni o’rnatish belgisi}
        Key: Word {Kalitni himoyalash shifri}
    end;
```

Bu tiplashtirilgan konstantani yaratish jaroyonida kompilyator HF maydonga False qiymatni joylashtiriladi. Ishga yurguzitilgan programma shu maydonni tekshiradi. Birinchi ishga yurgizishga HF=False va programma Save protsedurasini ishga da’vat etadi. Natijada faylda sarlovha va uzunlikning etalon ko’rinishi saqlanadi. Shu bilan birgalikda programma faylining HF maydoniga True qiymati kiritiladi. Shuning uchun har bir keyingi murojat qilishda Save protsedurasining o’rnida CheckFile protsedurasi ishga da’vat rtiladi va shu protsedura faylni tekshirish ishini bajaradi.

Tekshirish jarayonida fayl sarlovhasining etalon ko’rinishidan ozgina bo’lsa ham o’zgarishi sezilsa programma bu haqda, ya’ni EXE-faylni zararlanganligi haqida ma’lumot baradi. Bundan tashqari, buzulgan sarlovha va fayl uzunliklarini tiklashni taklif etadi.

Zarar ko’rgan faylni tiklash algoritmi quyidagi qadamlardan iborat.

1-qadam. Zararlangan fayl VIR kengaymali faylga nusxalanadi. Buning asosiy maqsadikerek bo'ganda tiklash jarayonini takrorlash va keyinchalik virusni tahlil qilish va yo'qotish uchun nushalashdir.

2-qadam. Zararlangan programma ishga yrgizish adrisini tahlil qilish. Agar virus faylni bosh qismida yaki o'rtasida o'rnashsa EXE-faylini tiklash murakkablashadi.

3-qadam. Virusni o'rnatish joyini aniqlash. Agar virus fayl oxrida joylashgan bo'lsa tiklanishi kerak bo'lgan faylga etalon sarlovha va faylning zararlangan qismi, ya'ni faylniing bosh qismidan (Tabloff+Relolnt*4 baytdan boshlab FileSize-HL baytgacha) boshlab tiklash va kiritish ishlari bajariladi.

Tekshirish va himoyalash jarayoni mustahkam bo'lishi uchun **kalitni o'zini** ham himoyalash maqsadga muofiqdir. Bu ish kalitni shifrlash yordamida bajariladi. Shifrlash uchun tasodifiy sanlarni generatsiy qilish protsedurasi ishlatiladi[7]. Shularni hisobga olib Save orotsedurasiga qo'yidagi fragmetni kiritamiz.

```
{Kalitni shifrlash}
Randomize;
Head.key:=Random ($FFFF);
With Head, Head.HE do
For k:=1 to 14 do
   Hem [k]:=HE.Hem[k] xor Key;
```

Randomize protsedurasi Head.Key shifrini turli programmalarda turlicha bo'lishni ta'minlaydi. CheckFile protsedurasining qo'yidagi operatorlari kalitni ya'ni boshlangichholatiga o'tkazishni ijro etadi.

```
{Kalitni deshifirlash}
With H,H.HE do
   For  k:=1 to 14 do
Hem[k]:=Hem[k] xor Key;
```

Loyihalanayotgan antuvirus moduli (ya’ni Antivirus moduli) ni ishlatish uchun foydalanuvchi programmasiga **uses** direktivasida modul nomini ko’rsatish kifoyadir.

Modulda CheckVirushesult global o’zgaruvchi tasvirlangan bo’lib programmani tekshirish natijasida haqida etarlicha ma’lumot beradi. Bu o’zgaruvchining qiymatlari quyidagicha bo’lishi mumkin.

0; {zararlangan fakti mavjud emas}

1; {Programmani birinchi ishga yurgizilishi, himoyalash o’rnatilishi}

-1; {Virus faydalanivchi ruxsati bilan zararlantirildi, yo’qotildi.}

-2; {Virus avtomatik ravishda yo’qotildi}

-3; {Tekshirish /NOANII kaliti bilan tuxtatildi}

-4; Virus programma bosh qismida joylashgandir

CheckFile programmasining ishlash jarayonida qo’yidagi holatlarni hisobga olish kerak.

Protsedura ishi to’xtatiladi, agar NOANII kaliti ishlatilsa, /NOQUERU kaliti esa virusni avtomatik ravishda o’chiradi, /NOALARM kaliti virusni ogohlantiruvchi ma’lumot bermasdan o’chiradi, /NOCODY kaliti esa zararlangan faylni nusxalashni man’ etadi.

Antivirus modulni programmada qo’llasak va masalan, testanti.exe fayliga kompilyatsiya (tarjima) qilsak u holda **testanti** kamandasini birinchi marta qo’llaganda.

TESTANTI>EXE fayliga himoya kiritilgan degan ma’lumotnoma va har bir keyingi ishlatilishlardan esa,

TESTANTI>EXE fayliga virus topildi degan ma’lumotnoma chiqariladi.

Yuqoridagilarni hisobga olsak, maddani ishlatish programmasi qo’yidagi ko’rinishini oladi.

```
Program Testanti;
    Uses Antivirus;
    Begin
```

```
                Case CheckVirusResult of
                0: writeln (ParamStr(0),'faylida virus mavjud emas');
                1: writeln (ParamStr(0),'faylida himoya o'rnatilgan');
                -1: writeln ('Virus faydalanuvchi ruxsati bilan o'rnatiladi');
                -2: writeln ('Virus avtamatik ravishda o'rnatildi');
                -3: writeln ('Tekshirish /NOANTI kaliti bilan to'xtatildi');
                -4: writeln ('Virus fayl bosh qismida jaylashgan.
O'chirish mumkin emas');
                end;
end.
```

Biz hozirgacha yangidan loyihalanadigan EXE faylning ,,vaktsinatsiya (emlash)" qilish soda usulni tahlil qildik.Mavjud, ya'ni ishlab chiqarilgan EXE-programmani qanday himoyalash kerak? Degan savol to'g'ilishi tabiydir. Bu savalga javob berishning ikkita yo'li mavjuddir. Birinchisi, Mahsus programma yaratishda va bu programma faydalanuvchi talabiga ko'ra EXE-programmalarni tekshiradi. Ikkinchisi esa, himoyalinishi kerak bo'lgan programma oxiriga kichik hajmga ega bo'lgan kodni-,, virus fagini" qushish hisoblanadi. Antivirus-fag katta hajmga ega bo'lmagan assembler-programmadan iboratdir. Bunday fagni yaratish uchun EXE-programmalar juda yaxshi bilish va tahlil qilish kerakdir[8].

EXE-programma quyidagi tartibda ish boshlaydi.

1) Programmaviy segment prefiksi (PSP) tashkil qilinadi. Bu ish DOSning $26 funksiyasi yordamida bajariladi.
2) Xotiraning qandaydir lokal sahasida EXE-faylning 28 bayt bash qismi ko'chiriladi.
3) Faylningt yuklashi kerak bo'lganqismini o'lchovi (hajmi) LengExe=(PsgeCnt-1)*512+PartPag; formula yordamida aniqlanadi.
4) Yuklanishi kerak bo'lgan qism fayl siljishi SeekExe=HdrSize*16; yordamida belgilanadi.
5) Programmani joylashtirish segment adresi PSP dankeyin jaylashtiriladi.

6) Programmaning yuklash kerak bo’lganqismi xotiraning uzluksiz sohasiga o’qiladi.
7) Fayl ko’rsatgichi ko’chirma jadvalning bosh qismiga o’rnatiladi.
8) Ko’chirish elementlari (soni ReloCnt) uchun o’qish, absolyut adrislash va so’zlarni ajratib olish amallari bajariladi.
9) Programma oxirida keyin MinMem va MaxMem qiymatlari asosida xotira ajratiladi.
10) Registrlar kerakli qiymatlar bilan tulairilib programma ishga yurgiziladi.
    a) ES va DS registrlar segment qiymatlarini qabul qiladi.
    b) AX normal qiymati O_ga teng bo’ladi.
    c) SS=StaraSeg+Reloss;
       SP=Exesp
    d) CS=StartSeg+Relocs;

Fag bashqaruvchi o’ziga olgandan keyin AX va DS registirlar qiymatlarini saqlab DS o’zining o’zining ma’lumotlar segmenti qiymatini yuklaydi.

Bu usul va mexanizmlar yangin tashkil qilinadigan programmalar uchun qulayliklarni oddiy nomoish etish bilan ko’rsatish mumkin. Katta programmalar va COM_fayllarni himoyalash ancha murakkabdir. PSP-ni tekshirish asosida tashkil qilinadigan himoya usullari disk yuklovchi sektorlarga o’rnatiladigan virusraldan himoya qilishda yaxshi natija bermaydi[9]. Bu erda ham maxsus antivirus programmalarni ishlab chiqarish maqsadga muofiq hisoblanadi. Bunday programmalar BIOS ga murojat qilish va disk kontrollerlarning xususiyatlarni hisobga olish asosida tashkil qilinadi.

Shunday qilib, biz bu bo’limda antivirus programmalarni loyihalashning bir necha yullarini muhokama qildik. Yangi tashkil qilinadugan EXE-programmalarni ,,vaktsinatsiya” qilish usulini tahlil qildik, ,, emlash” programmasi unit tipidagi modul shaklida loyihalanadi.

Bu modul EXE-fayl tarkibiga kodlashtirilgan kalit ma’lumotni kiritadi. Programmani (EXE-faylni) keyingi har bir qullanilishida shu tekshiriladi.

Shu g’oyalar asosida tayyor (oldindan tashkil qilingan) EXE-programmalarni himoyalash usullari haqida ham batafsil ma’lumot berdik.

XULOSA

Bitiruv malakaviy ishida antivirus programmalar yaratish mexanizimlarini ishlab chiqish va ularni qo’llash masalalari muhokama qilingan. Bu masalani yechishda quyidagi natijalar va xulosalarga erishildi:

- kompyuter viruslarni sinflash printsiplari, turlari, hisoblash tizimining programma ta’minotiga va apparaturasiga ta’siri tahlil qilindi;
- ixtiyoriy hisoblash tizmi muhitida muhim hisoblanadigan EXE-fayllarni himoyalash masalasi tanlab olinadi;
- bajariladigan (exe - programmalarni) fayllarni himoyalash sodda va qulay usullardan biri – ularni ,,vaktsinatsiyalash” (emlash) bosqichlari aniqlandi;
- ,,emlash” usullarning asosida yotadigan mexanizmlar (fayl yaratilishida uning tarkibida kalitli ma’lumotlarni yozib qo’yish, keyingi murojatlarda tekshirib turish) o’rganildi va ularni qo’llash algoritmik ketma-katligi ajratildi;
- tizimli programmalashning masalalari (exe - fayllarni strukturasini o’rganish va o’zgartirish, uzulishlarni qo’llash, fayl boshlangich holatini tiklash, standart modullar yaratish) ni yechishning katma-katligi texnologok sxemalar shaklida tasvirlandi va programma ta’minoti yaratildi.

Ishning natijalarini, ya’ni ko’rsatmalar, texnologik yechimlar va xulosalarni dasturlovchilarni tayyorlashda o’quv qo’llanmalar va materiallar sifatida va raqobotbardosh programma mahsulatlarni ishlab chiqarishda qo’llash maqsadga muofiqdir.

ADABIYOTLAR

1. Безруков Н. Классификация компьютерных вирусов MS - DOS и методы защиты от них - М.: ИКЭ, 1990. - 48с.
2. Фролов А., Фролов Г. Осторожно: Коипьютерые вирусы. - М.: Диалог - МИФИ, 1996 - 256с.
3. Безруков Н. Компьютерные вирусы. - М.: Наука , 1996. - 316с.
4. Безруков Н. Компьютерные вирусология: Спровочное руководство. Киев: УСЭ. 1998. - 286с.
5. Фаронов В. Турбо Паслаль. 7.0. Практика программирования. Учебное носабие. – М.: Нолщж, 2001. - 432с.
6. Шерстюк Ф. Вирусы и антивирусы на компьюрере IBM PC . - М.: ИнфоАрт,1999. - С.119 - 138.
7. Фёдоров А. Особенности программирования на Borland Pascal. - Киев: Диалектика,1999. - 144с.
8. Богословский А. Системное программирование на асимблере для IBM – совместимых персональных компьютеров. - М.: Память 1992. - 91с.
9. Фаронов В. Практика Windows - програмировальния. - М.: Информпечать, 2004. - 247с.
10. http: //www.ziyonet.uz
11. http: //Vlibrary.freenet.uz
12. http: //betaplus.ru
13. http: //www.sjftdrom.ru
14. http: //www.softblog.ru

ILOVALAR

```
{Modul antivirusa}
Unit Antivir;
{Modul exe-fayllarni zararlanishini tekshiradi}
{Zararlanish fakti aniqlansa malumot beradi}
   Interface
     var checkVirusResult:integer;
     {Programma ishining natijasi:
       =0:virus mavjud emas;
       =1:faylga himoya qo'yildi;
       =-1:Virus foydalanuvchi ruxsati asosida  o'chirildi;
       =-2:Virus avtomatik tarzda o'chirildi;
       =-3:Tekshirish / noanti kaliti yordamida to'xtatildi;
       =-4:Virus fayl bosh qismida joylashgan, o'chirish mumkin emas}
Implementation
  Uses Dos,Crt;
  var fff:file;
  Procedure CheckFile;
    Type {exe-fayl sarlovhasi}
    HeadExeType=record case Byte of
0:(Sign:word;{Signatura 'mz'=$5A4D}
   PartPag:word;{Qismiy sektor}
   PageCnt:word;{Sektorlar soni}
   ReloCnt:word;{ko'chirish jadvaldagi elementlar soni}
   HdrSize:word;{Sarlavha uzunligi}
   MinMem:word;{Kucha minimal hajmi}
MaxMem:word;{kucha maksimal hajmi}
ReloSS:word;{SS segmenyi boshlang'ich qiymati}
```

```
        ExeSp:word;{Sp steki boshlang'ich qiymati}
        ChkSum:word;{fayl tekshirish summasi}
        ExeIp:word;{Programma ish boshlash nuqtasi}
        ReloCs:word;{CS segment boshlang'ich qiymati}
        TablOff:word;{Ko'chirish jadvali siljishi}
        Overlay :word;);{overlay nomeri}
        1: (Hem:array[1..14] of word)
        end;
        {Zarlanmagan fayl kaliti}
        KeyType=record
              HE:HeadExeType;{fayl sarlovhasi}
              HL:LongInt;{fayl boshlang'ich uzunligi}
              HF:Boolean;{kalit o'rnatish belgisi}
              Key:word;{Shifrlash kaliti}
         end; S8=String[8];
        const
           H:KeyType=
          (HE:(Hem:(0,0,0,0,0,0,0,0,0,0,0,0,0,0));HL:0;
           HF:false;
           Key:0);
          Key:array[1..4] of S8=('/noanti','/noquery','/noalarm','/nocopy');
         Var f,ff:file; NAME:STRING;
              Noanti, Noquery, NoAlarm, Nocopy:Boolean;
Offset:word;Head:KeyType;
        Procedure AntiAIDS;
           {Topilgan virusni o'chiradi}
           Var c:char; p:^byte; Delta:LongInt;
            Procedure WriteVirFile;
              {f faylni ff faylga ko'chiradi}
```

```
        Var K,L:LongInt; P:^Byte;
          Begin L:=FileSize(f);
          Repeat
            K:=MaxAvail;
            If k>65520 then k:=65520;
            if K>L then K:=L; getmem(p,k);
            Blockread(f,p^,k); Blockwrite(ff,p^,k);
            FreeMem(p,k); dec(L,K);
          Until L=0;
          End;
Procedure FileInit;
   {Zararlangan fayl nusxasini yaratadi}
   begin
     Name:=Copy(Name,1, pos('.',Name));
     Assign(ff,name+'VIR'); ReWrite(ff,1);
    {ff-fayl nusxasi, kengaymasi-VIR}
    Seek(f,0); WriteVirFile; Close(ff);
   End;
Const
   Txt1='Virus fayl bosh qismida joylashgan';
   Txt2='O`chirish mumkin emas!';
   Txt3=#13#10'Ixtiyoriy tugmachani bosing!';
     Begin{AntiAIDS}
       If not NoCopy then FileInit;
       If Head.HE.ReloCS>H.HE.ReloCS
       Then
         Begin {Virus fayl oxirida}
           {sarlavhani asl nusxasini tiklaymiz}
           Seek(f,0); blockWrite(f,H.HE,sizeof(H.HE));
```

```
            {uzunlikni asl nusxasini tiklaymiz }
            Seek(f,H.HL); truncate(f); {virusni kesish}
        If NoAlarm or NoQuery
        Then CheckVirusresult:=-2
        Else checkVirusresult:=-1
    End
 Else
   Begin
      If not NoAlarm
  Then
     Begin write(txt1,txt2,txt3);
        C:=Readkey;
        If c=#0 then c:=Readkey;
      End;checkVirusResult:=-4; {virus o'chirilmadi}
End
End; {AntiAIDS}
Procedure Check;
     {fayl sarlavhasini tekshiradi}
      Var k:byte; S:string[1];
         Begin CheckVirusresult:=0; {deshifrlash}
           With H,H.HE do
  For k:=1 to 14 do Hem[k]:=Hem[k] xor key;
K:=1;
While (CheckvirusResult=0) and (k<=14) do
Begin
If H.HE.HEM[k]<>Head.HE.HEM[k]
Then CheckVirusResult:=-1; inc(k);
End;
If CheckVirusresult=-1
```

```
Then
Begin {Zararlanish fakti aniqlandi}
If NoAlarm
Then AntiAIDS
Else
Begin writeln('fayl', Name,' zararlangan!'#7);
If NoQuery
Then AntiAIDS
Else
Begin write('virusni o`cherish(Y/N)?');
S:=''; readln(s);
If (s='') or (upcase(s[1])<>'N')
Then  AntiAIDS
Else checkvirusResult:=-3;
End
End
End
End; {chek}
Procedure save;
{fayl sarlavxasini HEAD kons. saqlaydi}
Var s:longint; r:registers; p:pointer;
Begin
 With head, head.he do
  Begin
  HL:=filesize(f); {fayl uzunligi}
HF:=true; r.AH:=$62; MSDOS(r) ;
P:=@H;
S:=round((DSeg-r.BX-16+HdrSize)*16.0)+ofs(p^);
End;
```

```
Seek(f,s); {kalitni shifrlaymiz}
Randomize; Head.key:=random($FFFF);
With head,head.he do
For s:=1 to 14 do hem[s]:=HE.Hem[s] xor Key;
{fayilga kalitni kiritamiz}
BlockWrite(f, Head, SizeOf(H));
checkVirusResult:=1;
end;
Function upString(s:string):string;
{simvollarni yuqori registrga o'tkazish}
Var k:byte;
Begin
For k:=1 to length(s) do s[k]:=upcase(s[k]);
Upstring:=s
End;
Procedure Getparams;
Var k,j:byte;
Begin
NoAnti:=false; noQuery:=False;
noAlarm:=false; nocopy:=false;
for K:=1 to paramcount do
 for j:=1 to 4 do
if upstring(paramstr(k))=key[j]
then
case j of
1: Noanti:=False;  2: Noquery:=False;
3:Noalarm:=False; 4:Nocopy:=False;
End;
End; {GetParams}
```

```
Begin {ChekFile}
Name:=Paramstr(0); {fayl to'liq nomi}
GetParams;
If noAnti
Then CheckVirusResult:=-3
Else
Begin assign(f,name); reset(f,1);
Blockread(f,head,sizeof(h));
If H.HF
Then Check {sarlavxa tekshiriladi}
Else Save; {sarlavxa saqlanadi}
Close(f);
End
End; {chekfile}
Begin checkfile end.
```

```
Program UseAnti;
   Uses AntiVir;
   Begin
    Case CheckVirusResult of
      0: writeln(ParamStr(0),' fayilda virus topilmadi.');
      1: writeln(ParamStr(0),' fayil ximoya qilindi.');
     -1: writeln('virus foydalanuvchi ruxsati bilan o`chirildi');
     -2: writeln('virus avtomatik ravishda o`chirildi.');
     -3: writeln('tekshirish /NoAnti kaliti yordamida to`xtatildi.');
     -4: writeln('virus fayl bbish qismida o'chirish mumkin emas!');
    End
   End.
```