12 ma’ruza **Fan bo‘yicha qo‘shimcha adabiyotlar bilan ishlash.**
(Algoritmlash va dasturlash. Python, Java, C++).

Kalit so‘zlar: *algoritm, algoritmlash, dasturlash, Python dasturlash tili, Java, C++ dasturlash tili.*

Axborot texnologiyalarini rivojlanishining asosiy bo‘g‘ini bu algoritmlash jarayonidir. Biror bir masala yoki muommoni elektron hisoblash mashinasida hisoblash uchun, albatta, berilgan masalani matematik modeli, algoritmi va biror bir algoritmlash tili asosida dasturini yaratish kerak bo‘ladi. Muommo yoki masalani matematik modelini tuzish deganda berilgan masalani matematik tushunchalar bilan ifodalanishiga aytiladi. Muommo yoki masalani yechish qadamlari albatta algoritm bilan bog‘liqdir. Algoritm so‘zi va tushunchasi IX asrda yashab ijod etgan buyuk vatandoshimiz alloma Muhammad al-Xorazmiy nomi bilan uzviy bog‘liq. Algoritm so‘zi Al-Xorazmiy nomini Yevropa olimlari tomonidan o‘zgacha talaffuz qilinishidan yuzaga kelgan. Al-Xorazmiy birinchi bo‘lib o‘nlik sanoq sistemasining tamoyillarini va undagi to‘rtta amallarni bajarish qoidalarini asoslab bergan. Masala yoki muommoni hal etish uchun yuqorida keltirib o‘tilgandek, uni algoritmini tuzish kerak bo‘ladi.

Algoritm – bu biror bir masala yoki muommoni hal etish uchun kerak bo‘lgan chekli sondagi buyruqlar ketma- ketligi.

Python dasturlash tili va uning imkoniyatlari Python dasturlash tilining tarixi o‘tgan asrning 80-yillari oxirlarida boshlangan. Gido Van Rossum Python dasturlash tilini 1980-yillarda yaratgan va u til 1989 yil dekabrda Gollandiyadagi matematika va informatika laboratoriya markazida ishlab chiqilgan. Python istisno holatlarini ko‘rib chiqishga va Amoeba operatsion tizimiga ta’sir ko‘rsatishga qodir bo‘lgan ABC dasturlash tilining avlodi bo‘lgan. Van Rossum Pythonning asosiy muallifidir va u 2018 yilgacha tilni rivojlantirish bo‘yicha bir qancha ishlar olib borgan. Van Rossum tomonidan Python 1.2 versiyasi 1995 yili matematika va informatika laboratoriya markazida ishlayotgan paytda ishlab chiqarilgan. Python dasturlash tili mukammal darajada ishlab chiqilgan dasturlash tili bo‘lib u insoniyat oldidagi muammolarni hal qilish uchun juda mos til hisoblanadi. Python dasturlash tili,

dasturlash tillarining eng keng imkoniyat doirasiga ega hisoblanadi, bu dasturlash tili boshqa dasturiy vositalarni boshqarish va ularning tarkibiy qismlarini mustaqil boshqarishni amalga oshirdi. Aslida, Python ko‘p maqsadli dasturlash tili sifatida o‘rganilishi mumkin, bu dasturlash tili yordamida bir qancha jarayonlarni dasturlash imkoni yaratiladi. Python amaliy dasturiy maxsulotlar, web ilovalar va ilmiy dasturiy maxsulotlar yaratish imkonini beradi. Python tarkibida xotiradan foydalanish va ishlash talablari bo‘yicha cheklovlar mavjud emas, ya’ni imkoniyatlar shu qadar kattaki, boshqa dasturlash tillari kabi ma’lumotlarni e’lon qilish tabaqasi mavjud emas. Bu esa dastur yozish vaqti kamaytiradi va boshqarish qulayligini oshiradi.

Python dasturlash tilini bu qadar keng tarqalishining sababi juda katta miqdordagi yuqori sifatli tayyor bepul tarqatiladigan modullar mavjud va ularni siz dasturning istalgan joyidan foydalanishingiz mumkin. Tayyor modullardan foydalangan holda dasturni tuzish bir qancha optimal hisoblanadi. Dasturlash tili tarkibida fundamental algoritmlar, funksiyalar va modullar tayyor holatga keltirilgan, bunda faqatgina bu algoritm yoki funksiyalarga murojaat qilinsa yetarli siz faqat tegishli qismlarni tanlashingiz va ularni bir joyga to‘plashingiz kerak. Modullar har bir misolning boshida mavjud bo‘lgan import buyrug‘i yordamida biriktiriladi. Ko‘p ishlatiladigan modullar ikkita asosiy qismga bo‘lingan: - Python interpretatori bilan ta'minlangan standart kutubxonaning modullari (ushbu modullar doimo dastur bilan birga aktivlashadi); - Tashqi vazifa bajaruvchi modullar, bu modullar alohida dastur tarkibiga o‘rnatish orqali amalga oshiriladi.

Python dasturlash tilining web dasturlash sohasiga ham to‘g‘ridan to‘g‘ri qo‘llanilishi mumkin. Python an'anaviy ravishda oddiy va murakkab strukturali saytlarni yaratish uchun foydalaniladi. Bu jarayoning eng keng tarqalgan vositasi Django web platformasi hisoblanadi. Bu platforma orqali bir qancha keng ommaga tarqalgan mashhur tizimlar ishlab chiqilgan, jumladan, Instagram, Mozilla va hakoza. Django juda ko‘p turli xil funktsiyalarni, shu jumladan avtomatik ravishda hosil qilinadigan ma’lumotlar bazasini yaratish imkoniyatlarini taqdim etadi. Python dasturlash tili ko‘plab mashhur o‘yinlarni ishlab chiqish uchun ishlatiladi, 2000 -

yillarning birinchi yarmida Python dasturlash tili sivilizatsiyasida, to‘rtinchi o‘yinning ichki mantiqiy tuzilishini yozish uchun asosiy vosita sifatida ishlatildi.

Python dasturlash tilining matematik va ilmiy hisoblash jarayonlariga keng qo‘llanilish imkoniyati yaratilgan. Python umumiy maqsadlar uchun mo‘ljallangan til bo‘lib, u matematik paketlar bilan muvaffaqiyatli moslashuvni amalga oshiradi. Python dasturlash tilining asosiy xususiyati uning kengayish imkoniyatidir. Bu esa Python uchun nafaqat C va Fortrandagi algoritmlarning ko‘p sonli kutubxonalari yozildi va moslashtirildi. Python dasturlash tilining boshqa dasturiy vositalar va paketlardan foydalanish imkoniyati mavjud. Pythonni matematik paketga aylantirishning asosiy modullari ishlab chiqilgan. Python dasturlash tilining eng muhim afzalliklaridan biri shundaki, uning barcha amaliy kutubxonalari va qo‘shimcha maxsus modullarining rivojlanish muhiti bepul tarqatiladi. Bu esa Python dasturlash tilini rivojlantirish vositasi bo‘lishi mumkinligini anglatadi.

Python dasturlash tili dasturlashning quyidagi sohalarrida qo‘llaniladi:

- Tizimli dasturlash;
- Grafik interfeysli dasturlarni ishlab chiqish;
- Dinamik veb-saytlarni yaratish;
- Komponentlarning integratsiyasi;
- Ma'lumotlar bazalari bilan ishlash uchun dasturlarni ishlab chiqish;
- Ilmiy hisoblash uchun dasturlarni ishlab chiqish;
- O‘yinlarni rivojlantirish.

Arifmetik amallar. Berilganlarni qayta ishlash uchun dasturlash tillarida amallarning juda keng majmuasi aniqlangan. Amal - bu qandaydir harakat bo‘lib, u bitta (unar) yoki ikkita (binar) operandlar ustida bajariladi, hisob natijasi uning qaytaruvchi qiymati hisoblanadi. Tayanch arifmetik amallar dasturlash tilida quyidagicha yoziladi.

Dasturlash asoslarida arifmetik amallar matematikadagi amallarni yozilishi bir oz farq qiluvchi holatlari ham mavjud. Amallarni ishlash jarayoni tushunarli bo‘lishi uchun, ularni interaktiv rejimda sinab ko‘ramiz.

Matematik ifodasi	Python dasturlash tilida ifodasi	Izoh
+	+	qo‘shish
-	-	Ayirish
·	*	ko‘paytirish
:	/	bo‘lish
Qoldiqli bo‘lish	%	Qoldiqli bo‘lish
Butun bo‘lish	//	Butun bo‘lish
Darajaga ko‘tarish	**	Darajaga ko‘tarish

Misol. Arifmetik amallarni bajarilishi

```
Type "help", "copyright", "credits" or "license()" for more information.
>>> y=5
>>> x=2
>>> x+y
7
>>> x-y
-3
>>> x*y
10
>>> y/x
2.5
>>> y%x
1
>>> y//x
2
>>> y**x
25
>>>
>>> y=12.3
>>> x=7
>>> y//x
1.0
>>> y=12
>>> x=2.3
>>> y//2.3
5.0
>>>
```

Yuqoridagi misollarga e’tibor bersangiz // butun bo‘lish amali bo‘luvchi va bo‘linuvchi butun son bo‘lsa natija ham butun bo‘ladi. Agar bo‘luvchi va bo‘linuvchining kamida bittasi haqiqiy son bo‘lsa ham natija haqiqy bo‘ladi. Python dasturlash tilida amallarni funksiyalar orqai ham amalga oshirish imkoniyati mavjud.

Python dasturlash tilida ifodasi	Izoh
abs(x)	Modul
round(x)	Yaxlitlash
round(x, n)	n – xonagacha yaxlitlash
pow(x, y)	Darajaga ko‘tarish
divmod(x, y)	Butun va qoldiqli bo‘lish

Round(x) funksiya sonning butun qismigacha yaxlitlaydi, round(x,n) funksiyasi sonning n – xonasigacha yaxlitlaydi, pow(x,y)=x**y ga teng kuchli va a,b=divmod(x,y) funksiyasi bir vaqtda x ni y ga bo‘lib butun va qoldiq qismlarini oladi. Funksiyali amallarni ishlash jarayoni tushunarli bo‘lishi uchun, ularni interaktiv rejimda sinab ko‘ramiz.

Ta’minlash operatori.

Ma’lum bir ifodaning natijasi biror o‘zgaruvchiga ta’minlash uchun Python dasturlash tilida “=” belgisi bilan ishlatiladi va uning umumiy ko‘rinishi quyidagicha:

<o‘zgaruvchi>=<ifoda>;

Python dasturlash tilida taminlash operatori amallar yordamida ham ishlatiladi. Qo‘shish qiymat berish bilan (+=); ayirish qiymat berish bilan (-=); ko‘paytirish, qiymat berish bilan (*=); bo‘lish qiymat berish bilan (/=); bo‘lish qoldig‘ini olish qiymat berish bilan (%=) va boshqalar. Bu holatlarning umumiy ko‘rinishi:

<o‘zgaruvchi><amal>=<ifoda>;

s+=x; ning ma’nosi *s=s+x;*

Ta’minlash operatorini ishlash jarayoni tushunarli bo‘lishi uchun, ularni interaktiv rejimda sinab ko‘ramiz.

Mantiqiy amallar Mantiqiy turdagi o‘zgaruvchi xotiradan 1 bayt joy egallaydi va 0 (false, yolg‘on) yoki (true, rost) qiymat qabul qiladi. Mantiqiy tur o‘zgaruvchilar qiymatlar o‘rtasidagi munosabatlarni ifodalaydigan mulohazalarni rost (true) yoki yolg‘on (false) ekanligi tavsifida qo‘llaniladi va ular qabul qiladigan

qiymatlar matematik mantiq qonuniyatlariga asoslanadi. Mantiqiy o‘zgaruvchini quyidagicha faollashtiramiz.

<o‘zgaruvchi> = qiymat

Bu yerda qiymat True yoki False bo‘lishi mumkin. Taqqoslash amallari python dasturlash tilida quyidagi jadvalda berilgan ko‘rinishida bajariladi.

Python dasturlash tilida mantiqiy amallardan foydalanishda albatta quyidagilarga e’tibor bering: - O‘zgaruvchiga boshlang‘ich qiymatlarni berishda = belgisi oldidan va orqasidan bitta bo‘sh joy(probel) quying; - O‘zgaruvchiga boshlang‘ich qiymatlarni berishda True va False kabi yozilish kerak ya’ni birinchi harfi katta harflarda.

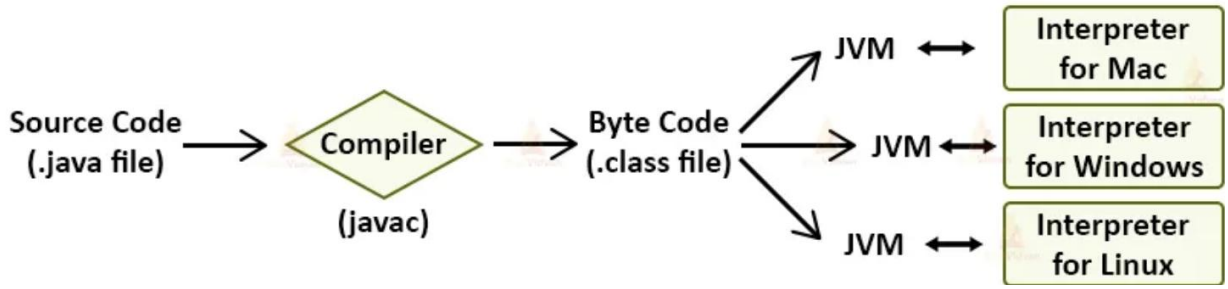
Java dasturlash tili.

Java – Sun Microsystems (*James Gosling va uning jamoasi*) tomonidan **1995**-yilda ishlab chiqarilgan dasturlash tili va hisoblash platformasi. Biroq hozirda u **Oracle** kompaniyasiga tegishlidir.

Java **obyektga yo’naltirilgan dasturlash tilidir**, yani ma’lumotlar va funksiyalar obektlar shaklida ifodalangan bo’lib, obektlar orqali amalga oshiriladi. Bu til xavfsizlikga ham katta e’tibor beradi. Javaning xavsizlik va portablity kabi xususiyatlarining asosiy kaliti shundaki, **Java compiler**i yozilgan kodni **executable code**ga o’tkazmaydi. Aksincha, uni **bytecode**ga o’giradi*(javac orqali)*. **Bytecode**ni ishga tushirish uchun shunchaki **JVM (Java Virtual Machine)** platformaga o’rnatilsa bas.

Java Virtual Machine o’zi nima?

JVM – Java bytecodelarini ishga tushirish uchun ishlatiladigan virtual mashina, platforma uchun tarjimon deb tushunsak ham bo’ladi.



C++ dasturlash tili

Hozirgi kunda juda ko‘p algoritmik tillar mavjud. Bular ichida Java va C++ dasturlash tillari universal tillar hisoblanib, boshqa tillarga qaraganda imkoniyatlari kengroqdir. So‘ngi yillarda Java va C++ dasturlash tillari juda takomillashib, tobora ommalashib bormoqda. Mazkur tillardagi vositalar zamonaviy kompyuter texnologiyasining hamma talablarini o‘z ichiga olgan va unda dastur tuzuvchi uchun ko‘pgina qulayliklar yaratilgan. C++ 1980 yillar boshida Bjarne Stroustrup tomonidan C tiliga asoslangan tarzda tuzildi. C++ juda ko’p qo’shimchalarni o’z ichiga olgan, lekin eng asosiysi u obyektlar bilan dasturlashga imkon beradi.

C++ dan tashqari boshqa ko'p obyektli dasturlshga yo’naltirilgan tillar paydo bo'ldi. Shulardan eng ko’zga tashlanadigani Xerox ning Palo Altoda joylashgan ilmiy-qidiruv markazida (PARC) tuzilgan Smalltalk dasturlash tilidir. Smalltalk da hamma narsa obyektlarga asoslangan.

C++ esa gibrid tildir. Unda C tiliga o’hshab strukturali dasturlash obyektlar bilan dasturlash mumkin. C++ funksiya va obyektlarning juda boy kutubhonasiga ega. Yani C++ dasturlash tilida dasturlashni o’rganish ikki qismga bo’linadi. Birinchisi bu C++ tilini o’zini o’rganish, ikkinchisi esa C++ ning standart kutubhonasidagi tayyor obyekt va funksiyalarni qo’llashni o’rganishdir.

C++ tiliga ko‘plab yangiliklar kiritilgan bo‘lib, tilning imkoniyati yanada kengaytirilgan. C++ dasturlash tili ham boshqa dasturlash tillari kabi o‘z alfavitiga va belgilariga ega.

- Tillarda mavjud alfavit va leksemalarga quyidagilar kiradi:
  1. Katta va kichik lotin alfaviti harflari;
  2. Raqamlar - 0,1,2,3,4,5,6,7,8,9;
  3. Maxsus belgilar: " {} | [] () + - / % \ ; ' : ? <=>_ ! & ~ # ^ . *
- Alfavit belgilaridan tilning leksemalari shakllantiriladi:
  - ✓ Identifikatorlar;
  - ✓ Kalit (xizmatchi yoki zahiralangan) so‘zlar;
  - ✓ O‘zgarmaslar;
  - ✓ Amallar belgilanishlari;
  - ✓ Ajratuvchilar.

Bu tillarda tuzilgan dasturlarda izohlar istalgan joyda berilishi mumkin. Ular satriy va blokli ko‘rinishlarda bo’ladi. Satriy izohlar uchun “//”, blokli izohlar uchun “/*”, “*/” belgilari ishlatiladi.

C++ dasturlash tilida dastur quyidagi tarkibda tashkil topadi: Direktivalar – funksiyalar kutubxonasini chaqirish. Ular maxsus include katalogida joylashgan va .h kengaytmali fayllar bo'ladi. C++ tilida masalaning qo'yilishiga qarab kerakli kutubxonalar chaqiriladi. Bus esa dasturning xotirada egallaydigan joyini minimallashtiradi. Masalan, ma'lumotlarni kiritish-chiqarish proseduralari uchun: #include tizimdan chaqirish #include "stdio.h" joriy katalogdan chaqirish. C++ dasturlash tili bilan ishlovchi eng sodda dasturlar Dev C++ va CodeBlocks dasturlaridir. Ularning tarkibida 300 dan ortiq kutubxonalar mavjud. Eng ko'p ishlatiladigan kutubxonalar quyidagilar:

```
#include<iostream.h>,
```



```
#include <math.h>
#include <conio.h>
#include <graphics.h>
#include <memory.h> va boshqalar
```

Makrolar (#define) – dastur bajarilishi davomida o'zgaruvchi ko'rsatilgan qiymatni qabul qilishi uchun (const). Unda makroning nomi va qiymati ko'rsatiladi.

Masalan,

```
#define pi 3.1415
#define x 556
#define s[100]
#define M x*x*x
```

main () funksiyasi– asosiy degan ma'noni anglatadi. Bu funksiya "{" belgisidan boshlanadi va dasturning asosini tashkil etuvchi o'zgaruvchilarning toifalari ko'rsatiladi. Dastur "}" belgisi bilan yakunlanishi shart. Agar dasturda qism dasturlardan foydalanilayotgan bo'lsa, ularning nomlari va haqiqiqy parametrlari keltiriladi. So'ngra dasturning asosiy buyruqlari yoziladi. Agar buyruqlar murakkab bo'lsas, ular alohida "{}" belgilari orasiga olingan bo'lishi kerak.

C++ tilida dasturning asosi bo’lmish buyruqlar kichik harflar bilan yoziladi. Buyruqlar nuqta-verguk bilan (;) yakunlanadi. Buyruqlar bir qator qilib yozilishi ham mumkin.

C++ dasturlash tilida dastur funksiya va funksiyalardan tashkil topadi. Agar dastur bir nechta funksiyalardan tashkil topgan bo’lsa, bir funksiyaning nomi main deb nomlanishi shart. Dastur aynan main funksiyasining birinchi operatoridan boshlab bajariladi.

C++ tilidagi dastur ko‘rinishini quyidagi misol yordamida keltirib o‘tamiz.

```
#include <iostream.h>                    // sarlavha faylni qo‘shish
int main  ()                             // bosh funksiya tavsifi
```



```
{                                        //  blok boshlanishi
cout << “Salom Dunyo! “\n”;              //   satrni chop etish
return 0;                                // funksiya qaytaradigan qiymat
}                                        //  blok tugashi
```

Dasturning 1-satrida #include direktivasi bo‘lib, dastur kodiga oqimli o‘qish/yozish funksiyalari va uning o‘zgaruvchilari e’loni joylashgan iostream.h sarlavha faylini qo‘shadi. Keyingi qatorlarda dasturning yagona, asosiy funksiyasi main() funksiyasi tavsifi keltirilgan. Shuni qayd etish kerakki, C++ dasturida albatta main() funksiyasi bo‘lishi shart va dastur shu funksiyani bajarish bilan o‘z ishini boshlaydi.

Dastur tanasida konsol rejimi (Consol – rejimi bu MS DOS oynasi ko’rinishiga o’xshash oyna bo’lib, unda foydalanuvchi dastur tuzuishda faqat dastur kodlari bilan ishlaydi. Graphic interface – rejimida esa faqat tilning kodlari bilangina emas muhitning menyulari, komponentalari bilan ham ishlashi mumkin bo’ladi) da belgilar ketma-ketligini oqimga chiqarish amali qo‘llanilgan. Ma’lumotlarni standart oqimga (ekranga) chiqarish uchun quyidagi format ishlatilgan:

cout << ;

Bu yerda sifatida o‘zgaruvchi yoki sintsksisi to‘g‘ri yozilgan va qandaydir qiymat qabul qiluvchi til ifodasi kelishi mumkin (keyinchalik, burchak qavs ichiga olingan o‘zbekcha satr ostini til tarkibiga kirmaydigan tushuncha deb qabul qilish kerak). cin << a; Ma’lumotlarni klaviatura yordamida kiritish buyrug’i bo’lib, u ham iostream.h kutubxonasi tarkibidagi funksiya hisoblanadi.

Nazariy savollar

1.Dasturlash tillarining sinflari va ularning tarkibi?

2.Quyi, o‘rta va yuqori darajali dasturlash tillarining vazifalari?

3.Quyi, o‘rta va yuqori darajali dasturlash tillari?

4.Yuqori darajali dasturlash tillarining bosqichlari?

5.Python dasturlash tilining imkoniyatlari?

6.Python dasturlash tilining boshqa dasturlash tillari bilan bog‘liqligi?

7.Python dasturlash tilining keng tarqalish sabablari?

8.Python dasturlash tilining amaliy, web va matematik paket dasturlarini yaratish imkoniyati?

9.Python dasturlash tilining sohalarda qo‘llanilishi?

10. Arifmetik amallarni Python tilida yozilishi?

11. Ta’minlash operatori va uning vazifasi?

12. Amal bilan beriladigan ta’minlash operatorini turllari va ularni Python tilida yozilishi?

13. Taqqoslash amallari va ularni Python tilida yozilishi?

14. Mantiqiy amal?

15. Inkor amali va uning Python tilida yozilishi;

16. Konyunksiya amali va uning Python tilida yozilishi?

17. Dizyunksiya amali va uning Python tilida yozilishi?

Foydalanilgan adabiyotlar ro‘yxati

1. Стенли Липпман. Язык программирование C++. Базовой курс. Вильямс - М.: 2014.

2. Сидхарма Рао. Освой самостоятельно C++ за 21 день. Вильямс - М.: 2013.

3. Никита Культин. Microsoft Visual C++ в задачах и примерах. БХВПетербург - Петербург., 2010.

4. Б. Страуструп. Язык программирования C++. Специальное издание.- М.:ООО «Бином-Пресс», 2006. - 1104 с.

5. Ноутон П., Шилдт Г. Java 2 в подлиннике.: Пер. с англ. - СПб.:БХВ-Петербург, 2000. - 1072 с.: ил.

6. Еккель Б. Философия Java. Библиотека программиста. 3-изд. - Спб: Питер, 2003.- 972 с.

7. Герберт Шилдт. Java. Руководство для начинающих. Вильямс. - М.: КУДИЦ-ОБРАЗ, 2012. - 624 с

8. Герберт Шилдт Java. 8-е издание. Исчерпывающее описание языка Java. Вильямс. - М.: КУДИЦ-ОБРАЗ, 2013. - 1104 с.

9. Кэти Сиерра, Берт Бейт Изучаем Java. Издательство: Эксмо, 2012, 720 с.

10. Д.Ю.Федоров. “Основы программирования на примере языка Python” Учебное пособие. Санкт-Петербург. 2-е изд., перераб. и доп. - М.: Издательство Юрайт, 2019 – 161 с.

11. Любанович Билл. “Простой Python. Современный программирования”. – СПб.: Питер, 2016 — 480 с.: ил. - (Серия «Бестселлеры O’Reilly»).

12. Васильев А.Н. “Python на примерах. Практический курс по программированию”. - СПб.: Наука и Техника, 2016 - 432 с.