O‘ZBEKISTON RESPUBLIKASI
OLIY TA’LIM, FAN VA INNOVATSIYALAR VAZIRLIGI

TURAN INTERNATIONAL UNIVERSITY

“TASDIQLAYMAN”
Turan International University rektori
_____ A.X.Najmitdinov
“24” _____12_____ 2025 y.

Ro‘yhatga olindi: _____B6_____
“24” _____12_____ 2025 y.

“60610800 – Axborot texnologiyalarining dasturiy ta’minoti” ta’lim yo‘nalishi fanlaridan yakuniy davlat attestatsiya sinovi

DASTURI

Namangan-2025

Annotatsiya

Mazkur dastur "Ta'lim to'g'risidagi" qonun, O'zbekiston Respublikasi Prezidentining 2019-yil 8-oktabrdagi "O'zbekiston Respublikasi oliy ta'lim tizimini 2030-yilgacha rivojlantirish konsepsiyasini tasdiqlash to'g'risida"gi PF-5847-sonli Farmoni, Vazirlar Mahkamasining 2024-yil 13-dekabrdagi "Oliy va kasbiy ta'lim tashkilotlari bitiruvchilarining yakuniy davlat attestatsiyasini tashkil etish chora-tadbirlari to'g'risida"gi 836-son qarori, O'zbekiston Respublikasi Prezidentining 2025-yil 28-apreldagi "Pedagog kadrlar tayyorlash tizimini yanada takomillashtirish chora-tadbirlari to'g'risida" PF-73-sonli Farmoni va Qarorlarida belgilangan vazifalar hamda olib borilayotgan islohotlar samarasini yanada oshirish, davlat va jamiyat rivojini yangi bosqichga ko'tarish, hayotning barcha sohalarini liberallashtirish, mamlakatimizni modernizatsiya qilish bo'yicha eng muhim ustuvor yo'nalishlar asosida belgilangan vazifalardan kelib chiqib "Turan International Universityda bakalavriat ta'lim yo'nalishi bitiruvchilarining yo'nalish fanlaridan yakuniy davlat attestatsiyalarini tashkil etish tartibi to'g'risida yo'riqnoma"ga asosan tayyorlandi.

Yo'nalish fanlardan tuzilgan sinov dasturi 60610800 – Axborot texnologiyalarining dasturiy ta'minoti ta'lim yo'nalishida o'qitiladigan fanlarning dasturi asos qilib olindi.

Dastur 5 ta bo'limdan iborat bo'lib, "Algoritmlash va dasturlash asoslari", "Operatsion tizimlar va tizimlarni dasturlash", "Ob'ektga yo'naltirilgan dasturlash texnologiyalari" va "Web-dasturlash va dizayn" kabi fanlarni o'z ichiga olib, unda ushbu ta'lim yo'nalishi bo'yicha bitiruvchiga quyilgan kasbiy bilim, ko'nikma va malaka darajasini baholash masalalari o'z aksini topgan.

Tuzuvchilar: **I. Xusainov**

H. Sobirjonov

Taqrizchi: **Z.Kodirov**
Turan Internaitonal University Biznes boshqaruvi kafedrasi mudiri, dotsenti

Dastur Turan International University "Biznes boshqaruvi" fakulteti yig'ilishida muhokama qilingan va tasdiqlash uchun Universitet kengashiga tavsiya etilgan 2025-yil "22" 12 dagi _____sonli majlis bayoni.

Dastur Turan International University kengashi tomonidan tasdiqlangan. 2025-yil "24" 12 dagi _____sonli majlis bayoni.

Kirish

Mamlakatimizda yuqori malakali kadrlar tayyorlash jarayonini sifat jihatidan yangi bosqichga ko'tarish, ish beruvchi korxona va tashkilotlarni ta'lim jarayoniga jalb etish hamda Turan International University 60610800 – Axborot texnologiyalarining dasturiy ta'minoti ta'lim yo'nalishi bitiruvchilarining yakuniy davlat attestatsiyasini o'tkazish tartibini takomillashtirish maqsadida ushbu Dastur ishlab chiqildi.

Mazkur dastur 60610800 – Axborot texnologiyalarining dasturiy ta'minoti ta'lim yo'nalishi bitiruvchilarining yakuniy davlat attestatsiyasini (YDA) tashkil etish va o'tkazish tartibini belgilaydi hamda bitiruvchilarning egallagan nazariy bilimlari, amaliy ko'nikmalari va kasbiy kompetensiyalarini baholash mezonlarini aniqlaydi.

Yakuniy davlat attestatsiyasining asosiy maqsadi – talabalar tomonidan ta'lim yo'nalishi bo'yicha davlat ta'lim standartlari va o'quv dasturlarida nazarda tutilgan bilim, malaka va ko'nikmalarning o'zlashtirilganlik darajasini aniqlash, ularning mustaqil kasbiy faoliyat yuritishga tayyorgarligini baholashdan iborat.

Attestatsiya jarayoni amaldagi normativ-huquqiy hujjatlar, oliy ta'limga oid davlat ta'lim standartlari, o'quv rejalari hamda tegishli tartib-qoidalarga muvofiq ravishda tashkil etiladi. Yakuniy davlat attestatsiyasi ochiqlik, xolislik va shaffoflik tamoyillari asosida o'tkaziladi.

Mazkur dasturda yakuniy attestatsiyaning shakllari, baholash mezonlari, davlat attestatsiya komissiyasining vazifalari, bitiruvchilarning huquq va majburiyatlari, shuningdek natijalarni rasmiylashtirish tartibi bayon etiladi.

I.Yakuniy davlat attestatsiyasi qo'yilgan umumiy qoidalar

1.Umumiy tushunchalar:

yakuniy davlat attestatsiyasi (keyingi o'rinlarda – attestatsiya) – bakalavriat ta'lim yo'nalishi va magistratura mutaxassisligi bitiruvchilarining bilimi, ko'nikma va malaka darajasini baholash jarayoni.

yakuniy davlat attestatsiya komissiyasi – bakalavriat ta'lim yo'nalishi va magistratura mutaxassisligi bitiruvchilarining bilimi, ko'nikma va malaka darajasini baholovchi soha mutaxassislaridan iborat komissiya.

2. Attestatsiya oliy ta'lim tashkilotidagi har bir bakalavriat ta'lim yo'nalishi va magistratura mutaxassisligi bo'yicha tashkil etiladigan komissiya tomonidan ta'lim yo'nalishlari va mutaxassisliklarning xususiyatidan kelib chiqib, oliy ta'lim tashkilotida yoki korxonada (tashkilotda) o'tkazilishi mumkin.

3. Attestatsiya quyidagi shakllarda o'tkazilishi mumkin:

- maxsus (majburiy) fanlar (kasbiy-amaliy fanlar) bo'yicha sinov;
- bitiruv malakaviy ishi himoyasi;
- diplom loyihasi himoyasi;
- magistrlik dissertatsiyasi himoyasi;
- ta'lim yo'nalishi va mutaxassislikka mos "startap" loyihasi hamda ratsionalizatorlik g'oyasi taqdimoti.

4. Ta'lim yo'nalishi va mutaxassislik bo'yicha maxsus (majburiy) fanlardan (kasbiy-amaliy) attestatsiya fanlar mazmuniga qo'yiladigan talablar bilan bir qatorda, malaka talablarida nazarda tutilgan bilim, ko'nikma va malaka darajasini aniqlashga yo'naltiriladi.

II.Komissiya faoliyatini tashkil etish tartibi

5. Komissiya o'z faoliyatini Oliy ta'lim tashkilotlari bitiruvchilarining yakuniy davlat attestatsiyasini o'tkazish tartibi to'g'risida Nizom talablariga muvofiq amalga oshiradi.

6. Komissiya raisligiga nomzodlar ro'yxati tegishli kafedra mudiri tomonidan shakllantiriladi hamda tasdiqlash uchun oliy ta'lim tashkiloti rektoriga (filial direktoriga) taqdim etiladi.

7. Komissiya raislari oliy ta'lim tashkilotining idoraviy mansubligidan qat'i nazar, ushbu ta'lim tashkilotida faoliyat yuritmayotgan olimlar, ishlab chiqarish sohasidagi tajribali mutaxassislar orasidan tegishli oliy ta'lim tashkiloti rektorining (filial direktorining) buyrug'i bilan bir kalendar yili muddatiga tasdiqlanadi.

Komissiyaning tarkibi oliy ta'lim tashkiloti rektori (filial direktori) tomonidan bitiruv semestri boshlanishidan kamida bir oy oldin tasdiqlanadi.

Komissiya kamida 5 nafar professor-o'qituvchi va mutaxassislardan iborat tarkibda shakllantiriladi hamda uning uchdan ikki qismi qatnashganda, vakolatli hisoblanadi.

Bunda komissiya tarkibi mutaxassis tayyorlovchi oliy ta'lim tashkilotining ilmiy-pedagog kadrlari (attestatsiya komissiyasi umumiy tarkibining 40 foizigacha), shuningdek, kadrlar buyurtmachisi bo'lgan korxonalarning (tashkilotlarning) yuqori malakali mutaxassislari, turdosh oliy ta'lim tashkilotlarining yetakchi professor-o'qituvchi va ilmiy xodimlari, sohaning ilg'or mutaxassislari va olimlaridan iborat bo'ladi.

8. Quyidagilar komissiyaning asosiy vazifalari hisoblanadi:

- bitiruvchining bilim, ko'nikma va malaka darajasining tegishli ta'lim yo'nalishi va mutaxassislik malaka talablariga muvofiqligini kompleks baholash;

- attestatsiya natijalari bo'yicha bitiruvchilarga tegishli akademik darajasini berish bo'yicha qaror qabul qilish;

- faoliyat natijalarini tahlil qilish va ular asosida oliy ma'lumotli kadrlar tayyorlashni takomillashtirish bo'yicha takliflar ishlab chiqish.

9. Attestatsiyada oliy ta'lim tashkilotining Vasiylik (Kuzatuv, Boshqaruv) kengashi va fuqarolarning o'zini-o'zi boshqarish organlari vakillari, bitiruvchilarning ota-onalari yoki ularning qonuniy vakillari hamda kadrlar buyurtmachilari jamoatchilik asosida kuzatuvchi sifatida qatnashishi mumkin.

Attestatsiya jarayoni onlayn rejimda ovozli videotranslyatsiya qilinadi hamda ushbu videotasvirlar bir yil davomida tegishli kafedrada saqlanadi.

III. Attestatsiyani o'tkazish tartibi

10. Attestatsiyaga tegishli fanlarni o'zlashtirgan, belgilangan kreditlarni to'plagan hamda to'lov-kontrakt shartlarini to'liq bajargan bitiruvchilar qo'yiladi.

Bitiruvchilar o'quv yuklamalarini to'liq bajargandan so'ng attestatsiyadan o'tishi majburiy hisoblanadi.

11. Bitiruvchilarning maxsus (majburiy) fanlar (kasbiy-amaliy) bo'yicha attestatsiyasi, diplom loyihasi (bitiruv malakaviy ishi) yoki magistrlik dissertatsiyasi himoyasi oliy ta'lim tashkiloti yoxud korxonada (tashkilotda) tashkil etiladi.

12. Oliy ta'lim tashkiloti tomonidan tajribali professor-o'qituvchilar hamda korxonalarning (tashkilotlarning) mutaxassislarini jalb etgan holda, attestatsiyaning dasturlari attestatsiya boshlanishidan kamida uch oy oldin ishlab chiqilib, oliy ta'lim tashkiloti Kengashi tomonidan tasdiqlanadi va oliy ta'lim tashkilotining rasmiy veb-saytida e'lon qilinadi.

13. Attestatsiya o'quv jarayoni jadvalida ko'rsatilgan muddatlarda, fakultet dekani yoki o'quv-uslubiy ishlarga mas'ul tarkibiy bo'linma (bo'lim, boshqarma) tomonidan tayyorlangan hamda oliy ta'lim tashkiloti o'quv ishlari bo'yicha prorektori tomonidan tasdiqlangan attestatsiya jadvali asosida o'tkaziladi.

14. Oliy ta'lim tashkiloti Kengashining qarori bilan bitiruvchi uchun belgilangan attestatsiyaning shakli bir o'quv yili davomida o'zgartirilishiga yo'l qo'yilmaydi.

15. Attestatsiya boshlanishidan ikki hafta oldin komissiya raisi bilan oliy ta'lim tashkiloti rektori (filial direktori) komissiyaning tayyorgarlik darajasi, komissiya raisi va a'zolarining vazifalari, bitiruvchilarga qo'yilayotgan talablar, tartib-intizom masalalari yuzasidan yig'ilish o'tkazadi.

16. Attestatsiyada plagiatga yo'l qo'yilmaydi. Bunday holat aniqlanganda, attestatsiya komissiyasi tomonidan bitiruvchiga tegishli attestatsiyadan qoniqarsiz baho qo'yiladi yoki qo'yilgan ijobiy baho bekor qilinadi.

Bitiruv malakaviy ishi hamda magistrlik dissertatsiyasi himoyadan ikki hafta oldin antiplagiat dasturi asosida tekshiriladi va plagiatga yo'l qo'yilganligi aniqlansa, takroran bir marotaba qayta tekshirishdan o'tkazishga ruxsat beriladi. Ikkinchi marotaba ham plagiat holatlari bartaraf etilmagan taqdirda, attestatsiya keyingi yilga qoldiriladi.

Bunda plagiat holatlari bartaraf etilgan taqdirda, bitiruvchiga attestatsiyada qatnashishga ruxsat beriladi.

17. Attestatsiyada bitiruvchilarning bilimi, ko'nikma va malakasi O'zbekiston Respublikasi Oliy ta'lim fan va innovatsiyalar vazirligi tomonidan belgilangan baholash mezonlari asosida baholanadi.

18. Attestatsiya natijalari attestatsiya o'tkazilgan kunning o'zida bitiruvchilarga e'lon qilinadi.

19. Attestatsiyada bitiruvchi qoniqarsiz baho bilan baholanganda bitiruvchiga diplom berilmaydi hamda attestatsiyani mazkur Nizomga muvofiq qayta topshiradi.

20. Komissiya bitiruvchilarga attestatsiyaning ijobiy natijalari asosida bakalavr (magistr) darajasini berish haqida qaror qabul qiladi.

Komissiyaning qarorlari ochiq ovoz berish orqali majlisda ishtirok etuvchilarning ko'pchilik ovozi bilan qabul qilinadi. Ovozlar teng bo'lgan hollarda, raisning ovozi hal qiluvchi hisoblanadi.

21. Oliy ta'lim tashkiloti bitiruvchisi attestatsiyadan muvaffaqiyatli o'tgandan so'ng bir oy muddat ichida unga muvofiq darajadagi oliy ma'lumot to'g'risidagi diplom va uning ilovasi beriladi.

O'quv rejasidagi barcha fanlarning kamida to'rtdan uch qismi bo'yicha "5" (a'lo) baho (90 - 100 ball) o'zlashtirish ko'rsatkichlariga erishgan, (boshqa fanlar bo'yicha o'zlashtirish ko'rsatkichi "4" (yaxshi) bahodan (70 - 89,9 ball) kam bo'lmagan), shuningdek, attestatsiyadan "5" (a'lo) baho (90 - 100 ball) ko'rsatkichiga ega bo'lgan bakalavriat bitiruvchisiga "imtiyozli" diplom beriladi.

22. Attestatsiyadan o'ta olmagan yoki attestatsiyaga qo'yilmagan bitiruvchi o'qish muddati tugagandan so'ng, uning arizasiga muvofiq attestatsiyani keyingi uch yil davomida qayta topshirish huquqiga ega.

Attestatsiyadan o'ta olmagan bitiruvchidan attestatsiyani qayta topshirish jarayonlarida to'lov undirilmaydi.

Attestatsiyadan o'ta olmagan bitiruvchi keyingi o'quv yili boshlanganidan keyin attestatsiyani qayta topshirishga ruxsat berishni so'rab oliy ta'lim tashkiloti rektori (filial direktori) nomiga ariza bilan murojaat qiladi. Bunda bitiruvchiga attestatsiyani keyingi har bir o'quv yilida (uch yil davomida) ikki martadan ortiq qayta topshirishga yo'l qo'yilmaydi.

Attestatsiyani qayta topshirishga oliy ta'lim tashkiloti rektorining (filial direktorining) buyrug'i bilan ruxsat beriladi.

IV. Attestatsiyani baholash natijalari yuzasidan shikoyat qilish tartibi

23. Attestatsiya natijasi bo'yicha olgan bahosidan norozi bo'lgan bitiruvchilar arizalarini ko'rib chiqish uchun oliy ta'lim tashkiloti rektorining (filial direktorining) buyrug'i bilan apellyatsiya komissiyasi tashkil etiladi.

Apellyatsiya komissiyasi oliy ta'lim tashkilotining ilmiy-pedagogik kadrlari (apellyatsiya komissiyasi umumiy tarkibining 40 foizigacha), shuningdek, tegishli soha bo'yicha korxonalarning (tashkilotlarning) malakali mutaxassislari, turdosh oliy ta'lim tashkilotlarining tajribali professor-o'qituvchilari va ilmiy xodimlari orasidan kamida 5 nafar a'zodan iborat tarkibda tuziladi.

Apellyatsiya komissiyasi tarkibiga joriy yilda komissiya ishida ishtirok etgan komissiya a'zolari kiritilmaydi.

24. Attestatsiyada qo'yilgan bahodan norozi bo'lgan bitiruvchilar attestatsiya baholari e'lon qilingan kundan boshlab uch ish kuni ichida apellyatsiya komissiyasiga murojaat qilish huquqiga ega.

V. Baholash mezonlari

60610800 – Axborot texnologiyalarining dasturiy ta'minoti ta'lim yo'nalishi bo'yicha Yakuniy Davlat attestatsiyasi yo'nalish fanlaridan test sinov savollari dasturi doirasida o'tkaziladi. Test savollari ta'lim yo'nalishining "Algoritmlash va dasturlash asoslari", "Operatsion tizimlar va tizimlarni dasturlash", "Ob'ektga yo'naltirilgan dasturlash texnologiyalari" va "Web-dasturlash va dizayn" kabi yo'nalish fanlarini ichiga olib 400 ta savoldan iborat bo'ladi. Test sinovini baholashning umumiy mezoni 0 baldan 100 balgacha tashkil etadi. Har bir savol maksimal 2 baldan baholanadi.

Savolnomadagi 50 ta test savolining javoblariga qo'yilgan ballar yig'indisi asosida bitiruvchi talabaning Yakuniy Davlat Attestatsiyasida yo'nalish fanlaridan to'plagan umumiy bali aniqlanadi.

№	Umumiy ball	Baho		YaDA topshirgan talabaning bilim darajasi
1	90-100	5	A'lo	Savolning mazmun va mohiyati to'liq ochib berilgan
2	70-89,9	4	Yaxshi	Javobda talabaning mustaqil mushohada yuritish qobiliyati seziladi
3	60-69,9	3	Qoniqarli	Savolga javob berishda masalaning mohiyatini tushunilganligi
4	0-59,9	2	Qoniqarsiz	Savol bo'yicha aniq tasavvurga ega emas

1-fan. "Algoritmlash va dasturlash asoslari" fanidan test topshiriqlari

1. Python dasturlash tilida o'zgaruvchi e'lon qilish uchun qaysi kalit so'z ishlatiladi.
2. Quyidagi qaysi ma'lumot turi Python tilida o'zgarmas hisoblanadi.
3. Python tilida funksiya qiymat qaytarmasa, u avtomatik ravishda qanday qiymat qaytaradi.
4. Quyidagi qaysi operator Python tilida mantiqiy "VA" operatsiyasini bajaradi.
5. String ma'lumot turida indekslash qaysi raqamdan boshlanadi.
6. Quyidagi kod qanday natija beradi. `x = 5; y = 2; print(x // y)`
7. Python tilida izoh qo'shish uchun qaysi belgi ishlatiladi.
8. Bool ma'lumot turi qanday qiymatlarni qabul qiladi.
9. Quyidagi qaysi operator qoldiqni hisoblash uchun ishlatiladi.
10. String o'zgaruvchisini butun songa o'tkazish uchun qaysi funksiya ishlatiladi.
11. List comprehension yordamida 0 dan 10 gacha bo'lgan juft sonlar ro'yxatini yaratish uchun to'g'ri sintaksis qaysi.
12. Quyidagi kod natijasi nima bo'ladi. `print(type([1, 2, 3]))`
13. List ro'yxatiga yangi element qo'shish uchun qaysi metod ishlatiladi.
14. Quyidagi kod qanday natija beradi. `my_list = [1, 2, 3]; print(my_list[-1])`
15. Tuple va list o'rtasidagi asosiy farq nima.
16. List ro'yxatidagi elementlar sonini aniqlash uchun qaysi funksiya ishlatiladi.
17. Quyidagi kod natijasi nima. `my_list = [1, 2, 3, 4]; print(my_list[1:3])`
18. List ro'yxatidan elementni faqat o'chirish uchun qaysi metod ishlatiladi.
19. Bo'sh ro'yxat yaratishning to'g'ri usuli qaysi.
20. Quyidagi kod qanday natija beradi. `print([1, 2] + [3, 4])`
21. Dictionary ma'lumot tuzilmasida kalit-qiymat juftligini qo'shishning to'g'ri usuli qaysi.
22. Quyidagi kod natijasi nima. `my_dict = {'a': 1, 'b': 2}; print(my_dict['a'])`
23. Dictionary'dan barcha kalitlarni olish uchun qaysi metod ishlatiladi.
24. Set to'plamining asosiy xususiyati nima.
25. Bo'sh set to'plamini yaratishning to'g'ri usuli qaysi.
26. Ikkita set to'plamining kesishmasi (intersection) ni topish uchun qaysi operator ishlatiladi.
27. Dictionary'da mavjud bo'lmagan kalitga murojaat qilganda xatolikdan qochish uchun qaysi metod ishlatiladi.
28. Quyidagi kod natijasi nima. `my_dict = {'x': 10}; print('x' in my_dict)`
29. Set to'plamiga element qo'shish uchun qaysi metod ishlatiladi.
30. Dictionary'dan barcha qiymatlarni olish uchun qaysi metod ishlatiladi.
31. Python tilida "agar" shartini ifodalash uchun qaysi kalit so'z ishlatiladi.
32. For siklida range(5) nechta iteratsiya bajaradi.
33. While sikli qachon to'xtaydi.
34. Quyidagi kod natijasi nima. `for i in range(2, 5): print(i)`
35. If-elif-else konstruksiyasida nechta elif bo'lishi mumkin.
36. Continue operatori nima vazifa bajaradi.
37. Quyidagi qaysi operator siklni to'liq to'xtatadi.
38. Nested loop (ichma-ich sikllar) da ichki sikl necha marta ishlaydi.
39. Pass operatori nima uchun ishlatiladi.
40. Quyidagi kod natijasi nima. `x = 10; print("Katta" if x > 5 else "Kichik")`
41. Funksiya yaratish uchun qaysi kalit so'z ishlatiladi.
42. Funksiya qiymat qaytarish uchun qaysi kalit so'z ishlatiladi.
43. Default parametrli funksiya yaratishning to'g'ri usuli qaysi.
44. Args (*args) nima uchun ishlatiladi.
45. Kwargs (**kwargs) qanday argumentlarni qabul qiladi.
46. Lambda funksiya nima.
47. Quyidagi lambda funksiya natijasi nima. `f = lambda x: x * 2; print(f(5))`
48. Global o'zgaruvchini funksiya ichida o'zgartirish uchun qaysi kalit so'z ishlatiladi.
49. Funksiya ichidagi local o'zgaruvchi funksiyadan tashqarida ko'rinma.
50. Rekursiv funksiya nima.

51. Faylni o'qish uchun ochishning to'g'ri usuli qaysi.
52. Faylga yozish rejimida ochish uchun qaysi rejim ishlatiladi.
53. Fayl bilan ishni tugatishdan keyin qaysi metod chaqirilishi kerak.
54. With operatori fayllar bilan ishlashda qanday afzallik beradi.
55. Faylning barcha qatorlarini ro'yxat sifatida o'qish uchun qaysi metod ishlatiladi.
56. Faylga qo'shimcha ma'lumot yozish (append) uchun qaysi rejim ishlatiladi.
57. Faylning mavjudligini tekshirish uchun qaysi modul ishlatiladi.
58. Binary rejimda fayl ochish uchun qaysi qo'shimcha belgi ishlatiladi.
59. Fayldan bitta qator o'qish uchun qaysi metod ishlatiladi.
60. Quyidagi kod nima qiladi. `with open("file.txt", "w") as f: f.write("Hello")`
61. Try-except bloki nima uchun ishlatiladi.
62. Quyidagi qaysi xato turi mavjud emasligi mumkin bo'lgan faylni ochishda yuzaga keladi.
63. Finally bloki qachon bajariladi.
64. O'z xato turini yaratish uchun qaysi sinfdan meros olish kerak.
65. Raise kalit so'zi nima uchun ishlatiladi.
66. ValueError xatosi qachon yuzaga keladi.
67. TypeError xatosi qachon yuzaga keladi.
68. Except blokida xato turini ko'rsatmasdan foydalanish mumkinmi.
69. IndexError xatosi qachon yuzaga keladi.
70. Multiple exception ushlashning to'g'ri sintaksisi qaysi.
71. Stack ma'lumot tuzilmasining asosiy prinsipi qaysi.
72. Queue ma'lumot tuzilmasini Python'da qaysi modul yordamida amalga oshirish mumkin.
73. Dictionaryda saqlash uchun vaqt murakkabligi (time complexity) O'rtacha holatda qancha.
74. List'da element qidirish operatsiyasining vaqt murakkabligi qancha.
75. Quyidagi kod qanday ma'lumot tuzilmasini yaratadi. `from collections import defaultdict; d = defaultdict(list)`
76. Tuple packing va unpacking nima.
77. List slicing operatsiyasi qanday ishlaydi. `my_list[start:end:step]`
78. Nested dictionary (ichma-ich lug'at) ga qanday murojaat qilinadi.
79. Set to'plamlarining birlashmasi (union) qanday topiladi.
80. Collections modulidagi Counter sinfi nima uchun ishlatiladi.
81. Binary search algoritmining vaqt murakkabligi qancha.
82. Bubble sort algoritmining eng yomon holatdagi vaqt murakkabligi qancha.
83. List comprehension odatdagi for siklga nisbatan qanday afzallik beradi.
84. Generator funksiya oddiy funksiyadan qanday farq qiladi.
85. Map funksiyasi nima qiladi.
86. Filter funksiyasi qanday ishlaydi.
87. Reduce funksiyasi (functools modulidan) nima uchun ishlatiladi.
88. Decorator nima.
89. List va tuple qaysi operatsiyada tezlik jihatidan farq qiladi.
90. Quyidagi kod murakkabligi qancha. `for i in range(n): for j in range(n): print(i, j)`
91. Sinfda konstruktor yaratish uchun qaysi maxsus metod ishlatiladi.
92. Private atribut yaratish uchun qaysi prefix ishlatiladi.
93. Inheritance (meros olish) nima.
94. Polymorphism (ko'p shaklilik) nima.
95. Encapsulation (inkapsulyatsiya) qanday amalga oshiriladi.
96. Super() funksiyasi nima uchun ishlatiladi.
97. Statik metod yaratish uchun qaysi dekorator ishlatiladi.
98. Class metod yaratish uchun qaysi dekorator ishlatiladi.
99. Property dekoratori nima uchun ishlatiladi.
100. Multiple inheritance (ko'p meros olish) qanday amalga oshiriladi.

2-fan. "Operatsion tizimlar va tizimlarni dasturlash" fanidan test topshiriqlari

1. Operatsion tizim nima.
2. Operatsion tizimning asosiy vazifasi nima.
3. Operatsion tizimdan foydalanishning asosiy maqsadi qaysi.
4. Qaysi operatsion tizim sifati muhim hisoblanadi.
5. Birinchi lampali mashinalar qachon yaratildi.
6. Paketli ishlov berish tizimi nima.
7. Multidasturlash nima.
8. CPU ning to'liq nomi nima.
9. RAM ning ma'nosi nima.
10. ROM nima.
11. Qaysi xotira o'chirilganda ma'lumotlarni yo'qotadi.
12. Bootstrap dasturi nima qiladi.
13. Fon Neyman arxitekturasi nima.
14. Ikkilamchi xotira qurilmasiga misol keltiring:
15. Operatsion tizimning asosiy funksiyasi qaysi.
16. GUI ning ma'nosi nima.
17. CLI nima.
18. Tizim chaqirig'i (system call) nima.
19. Qaysi qurilma kirish qurilmasi hisoblanadi.
20. Qaysi qurilma chiqish qurilmasi hisoblanadi.
21. DMA ning ma'nosi nima.
22. Uzilish (interrupt) nima.
23. Jarayon (process) nima.
24. Fayl nima.
25. Fayl tizimi nima uchun kerak.
26. Dasturni bajarish uchun u qayerda bo'lishi kerak.
27. BIOS nima.
28. Qaysi xotira turi eng tez.
29. Operatsion tizim qaysi komponentga mansub.

30. Foydalanuvchi interfeysi nima.
31. Xavfsizlik (security) nima uchun kerak.
32. Himoya (protection) nima.
33. Batching nima.
34. Real-time system nima.
35. Kompyuter shinasi (bus) nima.
36. Qurilma drayveri nima.
37. Bufer nima.
38. Spooling nima.
39. Multiprogramming afzalligi nima.
40. Time-sharing system nima.
41. Multidasturlashning asosiy maqsadi nima.
42. Kontekstni saqlash (context switching) nima.
43. Simmetrik ko'p protsessorli tizim (SMP) nima.
44. Assimmetrik ko'p protsessorli tizim nima.
45. Klasterli hisoblash tizimi nima.
46. Hard real-time system xususiyati nima.
47. Soft real-time system nima.
48. Taqsimlangan tizim (distributed system) nima.
49. Mijoz-server arxitekturasi nima.
50. Fayl-server nima qiladi.
51. Ma'lumotlar ombori serveri (database server) vazifasi nima.
52. Veb-server nima qiladi.
53. Proksi-server vazifasi nima.
54. Uzviy bog'langan tizim (tightly coupled) nima.
55. Kuchsiz bog'langan tizim (loosely coupled) nima.
56. Bulutli hisoblashlar (cloud computing) nima.
57. DRAM nima.
58. Hoisting nima.
59. Kernel (yadro) nima.
60. Tizim jarayoni (system process) nima.
61. Demon (daemon) nima.
62. Virtual xotira nima.
63. Sahifalash (paging) nima.
64. Segmentatsiya nima.
65. Troshing nima.
66. Rejalashtirish algoritmi (scheduling algorithm) nima.
67. FCFS (First-Come, First-Served) nima.
68. Round Robin algoritmi nima.
69. Prioritetli rejalashtirish nima.
70. Deadlock (bloklanish) nima.
71. Bloklanishning to'rtta zaruriy sharti qaysilar.
72. Banker's algorithm nima uchun ishlatiladi.
73. Semaphore nima.
74. Monitor (sinxronizatsiya vositasi) nima.
75. Kritik sektsiya (critical section) nima.
76. Race condition nima.
77. Interprocess communication (IPC) nima.
78. Pipe nima.
79. Shared memory nima.
80. Message passing nima.
81. Mutual exclusion nima.
82. Starvation nima.
83. Aging texnikasi nima uchun kerak.
84. Convoy effect nima.
85. Working set modeli nima.
86. Page fault nima.
87. LRU (Least Recently Used) algoritmi nima.
88. Belady's Anomaly nima.
89. TLB (Translation Lookaside Buffer) nima.
90. Fragmentation (parchalanish) nima.
91. Compaction nima.
92. Swapping nima.
93. Inode nima.
94. Directory (katalog) nima.
95. Hard link va soft link farqi nima.
96. RAID nima.
97. DMA (Direct Memory Access) afzalligi nima.
98. Interrupt handler nima.
99. System call va library call farqi nima.
100. Preemptive va non-preemptive rejalashtirish farqi nima.

3-fan. "Ob'ektga yo'naltirilgan dasturlash texnologiyalari" fanidan test topshiriqlari

1. Python tilida sinf (class) yaratish uchun qaysi kalit so'z ishlatiladi.
2. Quyidagi kodda xatolik qayerda. class Talaba: def __init__(self, ism): ism = ism
3. Inkapsulyatsiya nima uchun ishlatiladi.
4. Agar bola klass ota klassning metodini qayta yozsa, qaysi metod chaqiriladi.
5. Python tilida private atribut yaratish uchun qanday belgi ishlatiladi.
6. self kalit so'zi nimani anglatadi.
7. Konstruktor metodi qanday nomlanadi.
8. Meros nima.
9. a = Talaba() va b = a ifodalarida nechta ob'ekt yaratiladi.
10. Polimorfizm bu nima.
11. Klass atributi (class attribute) nima.
12. __init__ metodi nima.

13. Kompozitsiya nima.
14. Konstruktor ichida instance atributni yaratish uchun qaysi kalit so'z ishlatiladi.
15. OOP da "object" nima.
16. super() funksiyasi nima uchun ishlatiladi.
17. Getter va setter metodlari nima uchun ishlatiladi.
18. Klass atributini o'zgartirsak, barcha ob'ektlarda ham o'zgaradimi.
19. Protected atribut qanday belgilanadi.
20. class B(A) yozuvi nimani bildiradi.
21. Statik metod yaratish uchun qaysi dekorator ishlatiladi.
22. Instance atribut qayerda yaratiladi.
23. Klass metodi yaratish uchun qaysi dekorator ishlatiladi.
24. Ob'ekt yaratishda __init__ metodiga parametr berilmasa nima bo'ladi.
25. __str__ metodi nima uchun ishlatiladi.
26. Bola klass ota klassning metodlarini ishlatishi mumkinmi.
27. Instance atribut va klass atributi o'rtasidagi farq nima.
28. self.__yosh atributi qanday atribut.
29. isinstance() funksiyasi nima qiladi.
30. Instance metod ichida instance atributini o'zgartirish mumkinmi.
31. Method overriding nima.
32. __repr__ metodi nima uchun ishlatiladi.
33. Abstrakt klass nima.
34. Agar bola klassda ota klass bilan bir xil nomdagi atribut bo'lsa, qaysi ishlatiladi.
35. @property dekoratori nima uchun ishlatiladi.
36. Kompozitsiyada bir klass ikkinchi klassning ob'ektini qanday saqlaydi.
37. __del__ metodi nima.
38. Klass atributi qanday yaratiladi.
39. Encapsulation prinsipiga ko'ra qaysi to'g'ri.
40. Instance metod yaratishda birinchi parametr nima bo'lishi kerak.
41. Multiple inheritance nima.
42. Bola klass konstruktorida super().__init__() chaqirilsa nima bo'ladi.
43. hasattr() funksiyasi nima qiladi.
44. Kompozitsiya va meroslik o'rtasidagi asosiy farq nima.
45. __eq__ metodi nima uchun ishlatiladi.
46. __add__ metodi qaysi operatorni qayta yozish uchun ishlatiladi.
47. getattr() funksiyasi nima qiladi.
48. Polimorfizmda bir xil nomdagi metodlar turli klasslarda turlicha ishlashi nima deyiladi.
49. __len__ metodi nima qaytarishi kerak.
50. Protected atribut (self._x) tashqaridan ishlatilishi mumkinmi.
51. setattr() funksiyasi nima qiladi.
52. __str__ metodi qanday parametr qabul qiladi.
53. type() funksiyasi ob'ektga nisbatan nima qaytaradi.
54. @classmethod dekoratori bilan yaratilgan metod birinchi parametr sifatida nima qabul qiladi.
55. __call__ metodi ob'ektni qanday qiladi.
56. Bola klass ota klassning metodini override qilganda self parametrini yozish shartmi.
57. delattr() funksiyasi nima qiladi.
58. Ikki ob'ektni == operatori bilan solishtirganda default holda nima solishtiriladi.
59. __init__ va __new__ metodlari o'rtasidagi farq nima.
60. Statik metod ob'ekt yaratmasdan chaqirilishi mumkinmi.
61. Diamond problem qachon yuzaga keladi.
62. Ob'ekt yaratilgandan keyin unga yangi atribut qo'shish mumkinmi.
63. __dict__ atributi nima saqlaydi.
64. Ob'ekt yaratilganda qaysi metod avtomatik chaqiriladi.
65. MRO (Method Resolution Order) nima.
66. __gt__ metodi qaysi operatorni qayta yozish uchun ishlatiladi.
67. vars() funksiyasi nima qaytaradi.
68. "has-a" munosabati qaysi OOP prinsipiga mos keladi.
69. __slots__ nima uchun ishlatiladi.
70. Mutable klass atributi barcha ob'ektlar uchun umumiymi.
71. Metaclass nima.
72. Descriptor protokolida qaysi metodlar bo'lishi kerak.
73. Multiple inheritance da MRO (Method Resolution Order) qanday ishlaydi.
74. __enter__ va __exit__ metodlari qaysi protokolga tegishli.
75. Singleton pattern nima uchun ishlatiladi.
76. Factory pattern nima uchun ishlatiladi.
77. __getattribute__ va __getattr__ farqi nima.
78. Private atribut (self.__x) bola klassdan kirilishi mumkinmi.
79. Abstract Base Class yaratish uchun qaysi modul ishlatiladi.
80. @lru_cache dekoratori nima qiladi.
81. __slots__ ishlatilganda ob'ektga yangi atribut qo'shish mumkinmi.
82. Monkey patching nima.
83. Mutable default argument muammosi nima.
84. __init_subclass__ metodi nima uchun ishlatiladi.
85. Property setter orqali validatsiya qilish mumkinmi.
86. __prepare__ metodi qachon ishlatiladi.
87. Decorator pattern asosiy g'oyasi nima.
88. Dataclass nima uchun ishlatiladi.
89. __mro__ atributi nimani ko'rsatadi.
90. __set_name__ metodi nima qiladi.
91. Weak reference nima uchun ishlatiladi.
92. Metaclass da __call__ metodi qachon chaqiriladi.
93. Protocol class (typing modulidan) nima uchun ishlatiladi.

94. Immutable ob'ekt nima.
95. __class_getitem__ metodi nima uchun ishlatiladi.
96. Bola klass klass atributini o'zgartirsa, ota klassnikiga ta'sir qiladimi.
97. Cooperative multiple inheritance da super() qanday ishlaydi.
98. Descriptor asosan nima uchun ishlatiladi.
99. mixins nima va qachon ishlatiladi.
100. Lazy evaluation nima.

4-fan. "Web-dasturlash va dizayn" fanidan test topshiriqlari

1. Quyidagi HTML kodda qaysi qism noto'g'ri yozilgan. ```html <div class="container"> <p>Salom</paragraph> Dunyo </div> ```
2. Qaysi CSS qoida barcha <p> elementlarni qizil, 20px va qalin qiladi.
3. JavaScript'da o'zgaruvchi e'lon qilishning qaysi usuli noto'g'ri.
4. Quyidagi JavaScript kod bajarilganda natija qanday bo'ladi.```javascript let x = "5"; let y = 5; console.log(x + y); ```
5. CSS Flexbox bilan elementlarni to'liq markazga joylashtirish uchun qaysi kod kerak.
6. HTTP va HTTPS protokollari o'rtasida qanday asosiy farq bor.
7. Quyidagi CSS selectorlardan ID elementini tanlash uchun qaysi ishlatiladi.
8. JavaScript'da massiv (array) yaratishning qaysi usuli to'g'ri hisoblanadi.
9. Responsive dizayn uchun CSS Media Query qanday to'g'ri ishlatiladi.
10. HTML'da rasm (image) qo'yish uchun qaysi tag to'g'ri ishlatiladi.
11. CSS Grid Layout'da 3 ta teng kenglikdagi ustun yaratish uchun qaysi kod yoziladi.
12. JavaScript'da funksiya yaratish (declare) uchun qaysi usul to'g'ri.
13. Bootstrap framework'da qator (row) yaratish uchun qaysi class ishlatiladi.
14. DOM'da ID bo'yicha element topish uchun qaysi JavaScript metod ishlatiladi.
15. CSS'da elementni sahifada ko'rinmas qilish uchun qaysi xususiyat ishlatiladi.
16. Quyidagi JavaScript kod bajarilganda natija qanday chiqadi. ```javascript let arr = [1, 2, 3]; arr.push(4); console.log(arr.length); ```
17. HTML formada POST metodi yordamida ma'lumot yuborish uchun qaysi kod to'g'ri.
18. CSS'da matnni o'ng tomonga tekislash uchun qaysi xususiyat qo'llaniladi.
19. JavaScript'da string (matn) uzunligini aniqlash uchun qaysi usul ishlatiladi.
20. Client-Server arxitekturasida "client" atamasi nimani anglatadi.
21. SQL'da jadvalga yangi ma'lumot qo'shish uchun qaysi buyruq ishlatiladi.
22. REST API da ma'lumot olish (retrieve) uchun qaysi HTTP metod ishlatiladi.
23. CSS'da element atrofiga chegara (border) qo'yish uchun to'g'ri sintaksis qaysi.
24. JavaScript'da shart operatori (if statement) qanday to'g'ri yoziladi.
25. HTML'da havolani (link) yangi oynada ochish uchun qaysi atribut kerak.
26. CSS Box Model qanday qismlardan (layers) tashkil topgan.
27. JavaScript massividan ma'lum indeksdagi elementni o'chirish uchun qaysi metod ishlatiladi.
28. Bootstrap'da 4 ustunli grid yaratish uchun har ustun qanday class olishi kerak.
29. HTML5'da video fayl joylash uchun qaysi tag to'g'ri ishlatiladi.
30. CSS'da shrift o'lchamini (font size) o'zgartirish uchun qaysi xususiyat ishlatiladi.
31. Quyidagi kod bajarilgandan keyin console'da qanday natija ko'rsatiladi. ```javascript let obj = { a: 1, b: 2 }; let newObj = obj; newObj.a = 10; console.log(obj.a); ```
32. CSS Specificity (o'ziga xoslik) bo'yicha qaysi selector eng yuqori ustuvorlikka (priority) ega.
33. Quyidagi HTML formada validatsiya bo'yicha qaysi input majburiy emas (optional). ```html <form> <input type="text" required> <input type="email" required> <input type="tel"> <input type="submit"> </form> ```
34. JavaScript'da Hoisting mexanizmi haqida qaysi bayonot to'g'ri hisoblanadi.
35. CSS Flexbox'da flex-direction: column; xususiyati qanday ta'sir ko'rsatadi.
36. Quyidagi SQL so'rov bajarilganda qanday natija qaytaradi (return). ```sql SELECT * FROM users WHERE age > 25 AND city = 'Tashkent'; ```
37. JavaScript'da async/await sintaksisi qanday maqsadda ishlatiladi.
38. CORS (Cross-Origin Resource Sharing) xatoligi qachon yuzaga keladi.
39. Quyidagi CSS kod qo'llanilganda flex elementlar qanday joylashadi. ```css .parent { display: flex; justify-content: space-between;} .parent > .child { flex: 1;}```
40. SQL Injection hujumidan himoyalanish uchun qaysi usul eng xavfsiz hisoblanadi.
41. JavaScript'da Event Delegation pattern qanday ishlaydi va nima uchun foydali.
42. MongoDB ma'lumotlar bazasiga document qo'shish uchun qaysi metod qo'llaniladi.
43. REST API'da qaysi HTTP status kodi "Resurs topilmadi" (Not Found) xolatini bildiradi.
44. Quyidagi JavaScript kod qanday qiymat qaytaradi va nima uchun.```javascript console.log(typeof [1, 2, 3]); ```
45. CSS Grid Layout'da gap xususiyati qanday vazifani bajaradi.
46. LocalStorage va SessionStorage o'rtasida qanday asosiy farq mavjud.
47. Quyidagi CSS selector qaysi elementlarni tanlab oladi (select). ```css ul > li:first-child```
48. JavaScript'da Promise.all() metodi qanday ishlaydi va vazifasi nima.
49. XSS (Cross-Site Scripting) hujumidan himoyalanish uchun nima qilish kerak.
50. Quyidagi kod bajarilgandan keyin massiv "a" qanday bo'ladi. ```javascript let a = [1, 2, 3]; let b = a.map(x => x * 2); console.log(a); ```
51. CSS Position xususiyatining qaysi qiymati elementni document flow'dan olib tashlaydi.
52. MongoDB'da shartli qidiruv (conditional search) uchun qaysi sintaksis ishlatiladi.
53. JavaScript'da Closure nima va qanday vazifani bajaradi.
54. Bootstrap modallarni ochish va yopish uchun JavaScript kerakmi.
55. RESTful API'da PUT va PATCH metodlari orasida qanday farq bor.
56. CSS Pseudo-class :nth-child(2n) qaysi elementlarni tanlaydi.
57. JavaScript'da == va === operatorlari orasida qanday farq mavjud.
58. AJAX so'rov yuborish uchun qaysi JavaScript API/metod ishlatiladi.
59. CSS Media Query'da "min-width: 768px" sharti nimani bildiradi.
60. MySQL'da INNER JOIN operatori qanday vazifa bajaradi.
61. JavaScript'da Array.reduce() metodi qanday ishlaydi va vazifasi nima.
62. CSS z-index xususiyati qachon ishlaydi va ta'sir qiladi.
63. Node.js loyihasida package.json fayli qanday vazifa bajaradi.
64. HTTP POST va GET metodlari orasida qanday asosiy farq bor.

65. JavaScript'da setTimeout() funksiyasi qanday ishlaydi va vazifasi nima.
66. CSS Flexbox'da align-self xususiyati qanday ishlaydi.
67. JSON.parse() va JSON.stringify() metodlari orasida qanday farq bor.
68. CSRF (Cross-Site Request Forgery) hujumidan qanday himoyalanish mumkin.
69. CSS Transform xususiyati bilan qanday transformatsiyalarni amalga oshirish mumkin.
70. MongoDB'da aggregate() metodi qanday vazifa bajaradi.
71. Quyidagi kod nimani console'ga chiqaradi va buning sababi nima. ```javascript for (var i = 0; i < 3; i++) { setTimeout(() => console.log(i), 1000); } ```
72. CSS Specificity bo'yicha qaysi selector eng yuqori ustuvorlikka (priority) ega. ```css 1. div p { color: red; } 2. .container p { color: blue; } 3. #main p { color: green; } 4. p { color: yellow; } ```
73. React'da Virtual DOM mexanizmi qanday ishlaydi.
74. Quyidagi SQL so'rovni optimizatsiya qilish mumkinmi va qanday. ```sql SELECT * FROM users WHERE name LIKE '%Ali%'; ```
75. JavaScript Event Loop mexanizmi qanday ishlaydi.
76. CSS'da quyidagi kod muammosi nima va uni qanday hal qilish mumkin. ```css .parent { height: 100%; } .child { height: 50%; } ```
77. Node.js'da "callback hell" muammosini qanday hal qilish mumkin.
78. Database Normalizatsiya nima va nima uchun kerak.
79. Quyidagi kod nimani chiqaradi va buning sababi nima. ```javascript const arr = [1, 2, 3]; arr[10] = 99; console.log(arr.length); console.log(arr[5]); ```
80. CSS Grid va Flexbox qachon qaysi birini ishlatish yaxshiroq.
81. JWT (JSON Web Token) qanday ishlaydi va qanday xavflari bor.
82. MongoDB'da Index'lar performance'ga qanday ta'sir qiladi.
83. Quyidagi JavaScript pattern qanday nomlanadi va nima uchun foydali. ```javascript (function() { var secret = "Ma'lumot"; // kod })();```
84. Progressive Web App (PWA) ning asosiy xususiyatlari qaysilar.
85. SQL'da Transaction nima va ACID printsiplari nimani anglatadi.
86. Quyidagi kod memory leak yaratadimi. Nima uchun. ```javascript function createElements() { const arr = []; for (let i = 0; i < 1000; i++) { arr.push(document.createElement('div'));} } setInterval(createElements, 1000); ```
87. REST API'da versiyalash (API versioning) uchun eng yaxshi usul qaysi.
88. CSS'da BEM (Block Element Modifier) metodologiyasi nima uchun kerak.
89. Webpack nima qiladi va nima uchun zamonaviy rivojlanishda kerak.
90. Quyidagi kod race condition yaratadimi va nima uchun. ```javascript let counter = 0; async function increment() { const temp = counter; await new Promise(resolve => setTimeout(resolve, 100)); counter = temp + 1; } Promise.all([increment(), increment()]); ```
91. GraphQL va REST API o'rtasidagi asosiy farqlar qaysilar.
92. CSS Custom Properties (CSS variables) brauzer mosligini qanday tekshirish mumkin.
93. NoSQL va SQL ma'lumotlar bazalari qachon qaysi birini tanlash kerak.
94. Server-Side Rendering (SSR) va Client-Side Rendering (CSR) farqi va foydalari.
95. Mikroservislar arxitekturasi (Microservices) nima va monolitdan farqi.
96. HTTP/2 ning HTTP/1.1 dan qanday afzalliklari bor.
97. OAuth 2.0 autentifikatsiya oqimi (flow) qanday ishlaydi.
98. Docker containerlar va Virtual Machinelar o'rtasida qanday farq bor.
99. Code Splitting va Lazy Loading nima va nima uchun muhim.
100. CI/CD (Continuous Integration/Continuous Deployment) nima va qanday amaliyotlar.

Tavsiya etilgan adabiyotlar ro'yxati

1. Operatsion tizimlar (O'quv qo'llanma) UR. Xamdamov, DJ.B.Sultonov, S.S.Parsiyev, U.M.Abdullayev-Toshkent-2021-436 b.
2. William Shotts. "The Linux Command Line: A Complete Introduction."
3. Richard Blum, Christine Bresnahan. "Linux Command Line and Shell Scripting Bible."
4. Javlon Abdullo "Mukammal dasturlash 2-kitob: JavaScript"2022-300b. ISBN 978-9943-6501-6-9
5. Javlon Abdullo "Mukammal dasturlash 1-kitob: HTML va CSS"2022-288b. ISBN 978-9943-6964-4-0
6. Maxarov T.A., Maxarov Q.T., Raximova M.A. "Web dasturlash. HTML, CSS, JavaScript. O'quv qo'llanma." – Toshkent: "NIF MSH", 2022, 180b.
7. Bertrand Meyer (2009). Touch of Class: Learning to Program Well with Objects and Contracts. Springer Science & Business Media. p. 329
8. Eric Matthes. "Python Crash Course: A Hands-On, Project-Based Introduction to Programming". No Starch Press, Inc. 2013.
9. Aripov M.M. Dasturlash asoslari. O'quv qo'llanma. "Tafakkur bo'stoni", Toshkent - 2015. – 240 b.
10. Mo'minov B.B. Dasturlash 2. Darslik. – T: "Nihol pritn", 2021. – 604 b.
11. SH.F. Madraximov "C++ tilida dasturlash asoslari". Toshkent NIF-MSH 2022.