

«М.Тынышпаев атындағы ALT университеті» АҚ
Энергетика және сандық технологиялар институты
Ақпараттық-коммуникациялық технологиялар кафедрасы



«6B06118 - Бағдарламалық инженерия»
білім беру бағдарламасы бойынша
АТТЕСТАТТАУ (КЕШЕНДІ) ЕМТИХАНЫНЫҢ БАҒДАРЛАМАСЫ

Алматы, 2025

"6B06118 - Бағдарламалық инженерия" білім беру бағдарламасының базалық және бейіндік пәндері бойынша аттестаттау (кешенді) емтиханының бағдарламасы КР Ғылым және жоғары білім министрінің 2022 жылғы 20 шілдедегі № 2 бүйрұғымен бекітілген МЖМБС-ға, тиісті үлгідегі білім беру ұйымдары қызметінің үлгілік қағидаларына және Қазақстан Республикасы Білім және ғылым министрінің 2018 жылғы 30 қазандығы № 595 бүйрұғымен бекітілген түрлері, жаңа редакцияда-КР Білім және ғылым министрінің міндеттін атқарушының 2021.12.29 № 614 бүйрұғымен, "6B06118 - Бағдарламалық инженерия" білім беру бағдарламасымен, пәндердің жұмыс оку бағдарламаларына сәйкес құрастырылған.

Оку бөлімі: күндізгі.

Бағдарлама ақпараттық-коммуникациялық технологиялар кафедра отырысында қаралды және талқыланды

№ 2 хаттама, 14 «қазан» 2025 жыл.

Ақпараттық-коммуникациялық технологиялар
кафедра менгерушісі



Касымова Д.Т.

Бағдарлама энергетика және сандық технологиялар институтының ОӘБ отырысында қаралды және мақұлданды

№ 2 хаттама, 20 «қазан» 2025 жыл.

Энергетика және сандық технологиялар
институтының ОӘБ төрағасы



Тойгожинова А.Ж.

Бағдарлама ОӘК отырысында қаралды және ОӘК бекітуге ұсынылды
№ 2 хаттама, 20 «қазан» 2025 жыл.

ОӘК төрағасы



Абдрешов Ш.А.

Мазмұны

1. Аттестаттау (кешенді) емтиханының мақсаты	4
2. Аттестаттау (кешенді) емтиханын өткізу регламенті	4
3. Білім алушылардың білімін бағалау критерийлері мен көрсеткіштері	5
4. Аттестаттау (кешенді) емтиханының мазмұны	7
5. Ұсынылатын әдебиеттер	11

1. Білім беру бағдарламасы бойынша аттестаттау (кешенді) емтиханының мақсаты

"6B06118 - Бағдарламалық инженерия" білім беру бағдарламасының базалық және бейіндік пәндері бойынша аттестаттау (кешенді) емтиханының мақсаты КР Ғылым және жоғары білім министрінің 2022 жылғы 20 шілдедегі № 2 бүйрұғымен бекітілген бітірушілердің даярлық деңгейінің мемлекеттік бакалавриат талаптарына сәйкестігін, қызметтің үлгілік қағидаларын айқындау болып табылады КР Білім және ғылым министрінің 2018 жылғы 30 қазандығы № 595 бүйрұғымен бекітілген Тиісті үлгідегі және түрдегі білім беру үйімдары жана редакцияда-КР Білім және ғылым министрінің міндетін атқарушының 2021.12.29 № 614 бүйрұғымен, "6B06118 - Бағдарламалық инженерия" білім беру бағдарламасының және пәндердің жұмыс оку бағдарламалары.

Аттестаттау (кешенді) емтиханының өткізу кезінде "6B06118 - Бағдарламалық инженерия" білім беру бағдарламасына сәйкес бітірушінің теориялық білімі де, практикалық дағдылары да тексеріледі.

2. Білім беру бағдарламасы бойынша аттестаттау (кешенді) емтиханының өткізу регламенті

Қорытынды аттестаттау ретінде аттестаттау (кешенді) емтиханы академиялық күнтізбеле сәйкес және ағымдағы оку жылына арналған оку процесінің кестесіне сәйкес өткізіледі.

Емтиханға толық оку курсын аяқтаған және оку жоспарында көзделген барлық алдыңғы аттестаттау сынақтарынан сәтті өткен адамдар жіберіледі. Білім беру бағдарламасы бойынша білім алушыларды қорытынды аттестаттаудан өткізу үшін "6B06118 - Бағдарламалық инженерия" аттестаттау комиссиясы (бұдан әрі - АК) құрылады.

АК төрағасы мен комиссияның дербес құрамын ALT университетінің Басқарма төрайымы-Ректоры бекітеді.

Аттестаттау комиссиясының құзыретіне:

- білім беру бағдарламаларының талаптарына белгіленген бітіруші кадрлардың теориялық және практикалық даярлығының сәйкестік деңгейін тексеру

- бітірушіге тиісті білім беру бағдарламасы бойынша бакалавр дәрежесін беру
- кадрларды даярлау сапасын одан әрі жақсартуға бағытталған ұсыныстар әзірлеу.

Базалық және бейіндеу пәндері бойынша аттестаттау (кешенді) емтиханының бағдарламасы емтихан өткізудің болжамды күніне дейін бір айдан кешіктірілмей білім алушылардың назарына жеткізіледі және Академияның сайтында жарияланады.

АК білім алушылардың жауаптарын талқылауды және түпкілікті бағалауды қорытынды бағаны – балдық-рейтингтік мәнде (1 – кесте-білім алушылардың білімін бағалау критерийлері мен көрсеткіштері) айқындаі отырып, жабық отырыста жүргізеді.

Аттестаттау (кешенді) емтиханының нәтижелері емтихан тапсырған күні білім алушыларға жеткізіледі.

Оң бағаны арттыру мақсатында аттестаттау (кешенді) емтиханын қайта тапсыруға жол берілмейді.

"Қанағаттанарлықсыз" деген баға алған аттестациялық (кешенді) емтиханды қайта тапсыруға қорытынды аттестаттаудың осы кезеңінде рұқсат етілмейді.

Қорытынды аттестаттау бойынша "қанағаттанарлықсыз" деген баға алған білім алушы Президент-Ректордың бүйрұғымен академиядан "білім беру бағдарламасының талаптарын орындаған: аттестациялық (кешенді) емтихан тапсырмаған" ретінде шығарылады.

Қорытынды аттестаттаудан өткен және білім беру бағдарламасын менгергенін растиған білім алушыға аттестаттау комиссиясының шешімімен "бакалавр" дәрежесі беріледі және қосымшасы бар диплом беріледі.

3. Білім алушылардың білімін бағалау критерийлері мен көрсеткіштері

Таблица 1

№	Әрпіктік жүйеде бағалау	балдардың сандық баламасы	% мазмұны	дастурлі жүйе бойынша бағалау	бағалау критерийлерінін көрсеткіштері
					6
1	A	4,0	95-100	әте жақсы	<ol style="list-style-type: none"> Жүйелердің даму перспективалары туралы түсінігі бар; Қосымша білімді көрсетеді; Теориялық білімді практикамен байланыстырады; Арнайы терминологияны еркін менгерген; Техникалық құралдардың себеп-салдарлық байланыстарын белгілейді; Болжам жасай алады; Қосымша сұрақтарға сенімді жауап береді.
2	A-	3,67	90-94	жақсы	<ol style="list-style-type: none"> Теорияның негіздерін ескере отырып, ол материалды әте жақсы көрсетеді; Толығымен жауап береді, өз бетінше қорытынды жасайды және жалпылаиды; Арнайы терминологияны жақсы біледі; Техникалық құралдардың себеп-салдарлық байланыстарын белгілейді; Қосымша сұрақтарға толық жауап береді.
3	B+	3,33	85-89	жақсы	<ol style="list-style-type: none"> Материалды жақсы көрсетеді; Толық жауап береді; өз бетінше қорытынды және жалпылау жасайды; Арнайы терминологияны менгерген; Материалды ұсыну кезінде логиканы біледі; Қосымша сұрақтарға жауап береді.
4	B	3,0	80-84	жақсы	<ol style="list-style-type: none"> Негізгі материалды біледі; Дәлелді мысалдар келтіреді; Жалпылау мен қорытынды жасайды; Арнайы терминологияда, презентация логикасында дәлсіздіктерге жол береді; Қосымша сұрақтарға жауап береді.
5	B-	2,67	75-79	жақсы	<ol style="list-style-type: none"> Негізгі материалды біледі, бірақ логикасыз түсініксіз жауап береді; Терминдерді қолдану кезінде дәлсіздіктерге жол береді; Қосымша сұрақтарға жауап беру кезінде қателіктер жібереді.
6	C+	2,33	70-74	жақсы	<ol style="list-style-type: none"> Тек теориялық білімнің негіздеріне ие; Қорытынды мен жалпылауды білмейді; Арнайы терминологияны толық пайдаланбайды; Қосымша және нақтылау сұрақтарына жауап береді.

7	C	2,0	65-69	қанағаттанарлық қанағаттанарлықсыз	1. Негізгі материал туралы толық білімі жоқ; 2. Ұсынылған материалдың логикалық байланысы жоқ. 3. Жауаптар фрагментті; 4. Қосымша сұрақтарға толық жауап берілмейді.
8	C-	1,67	60-64		1. Негізгі материал туралы толық білімі жоқ; 2. Дәлсіздіктерге жол береді, қорытынды, жалпылау жасай алмайды; 3. Жауаптар дәл емес және үзінді; 4. Қосымша сұрақтарға жауап беру кезінде қателіктер жібереді.
9	D+	1,33	55-59		1. Материалда әлсіз бағдарланған; 2. Сұраққа жауап беру логикасын білмейді; 3. Қосымша сұрақтарға жауап беру қыын.
10	D	1,0	50-54		1. Материалдың маңызды бөлігін білмейді; 2. Материалды үстірт көрсетеді; 3. Қосымша сұрақтарға жауап беру қыын;
11	FX	0,5	25-49		1.Мағынасын түсінбей ұсыну, үзік-үзік білімдер; 2. Қосымша сұрақтар ойлауды қыннатады; 3. Теорияның формальды түрде жатталған ережелері.
12	F	0	0-24		1. Жаттанды үзік-үзік білім; 2. Қосымша сұрақтарға жауап бере алмайды; 3. Сұрақтар мен есептердің мағынасын түсінбейді

4. "6B06118 - Бағдарламалық инженерия" білім беру бағдарламасы бойынша аттестаттау (кешенді) емтиханының мазмұны

Емтихан сұрақтарының тақырыбы базалық және бейіндік пәндер бойынша жұмыс оку бағдарламаларының циклдерінің таңдалған бөлімдеріне сәйкес келеді:

- 4.1. Бейіндік пән - «**Бағдарламаларды әзірлеудің аспаптық құралдары**»
- 4.2. Бейіндік пән - «**Компьютерлік модельдеу негіздері**»
- 4.3. Негізгі пән - «**Объектілі-бағдарланған программалау**»

4.1. Бейіндік пән - «БАҒДАРЛАМАЛАРДЫ ӘЗІРЛЕУДІҢ АСПАПТЫҚ ҚҰРАЛДАРЫ»

4.1.1. Бағдарламалық қамтамасыз етуді әзірлеудің аспаптық құралдарына кіріспе

Аспаптық құралдар ұғымы. Әзірлеу құралдарының жіктелуі (редакторлар, IDE, нұсқаларды бақылау жүйелері, тестілеу құралдары және т.б.). Бағдарламалық қамтамасыз етудің өмірлік цикліндегі құралдардың рөлі. Әзірлеу үдерістерін автоматтандыру. Қазіргі тенденциялардың шолу (DevOps, CI/CD, AI-құралдар).

4.1.2. Бағдарламаларды әзірлеу ортасы (IDE)

IDE архитектурасы мен компоненттері. Код редакторы: синтаксисті бояу, автотолықтыру, рефакторинг. Жобаны басқару консолі. Құрастыру және іске қосу құралдары. Плагиндер мен кеңейтімдер. Танымал IDE-лерді салыстыру (VS Code, PyCharm, IntelliJ IDEA, Eclipse).

Нұсқалаудың (версионирование) ұғымы. Орталықтандырылған және таратылған нұсқа бақылау жүйелері. Git-тің негізгі командалары (commit, push, pull, merge). Бөлімшелермен (branches) жұмыс істей. Қактығыстарды шешу. GitHub/GitLab/Bitbucket репозиторийлері. Командалық жұмыс және workflow (GitFlow).

4.1.3. Бағдарламаларды жөндеу (отладка) құралдары

Бағдарламалаудағы қателер түрлері (логикалық, синтаксистік, runtime-қателер). Debugger: тоқтау нүктелері, айнымалыларды қарастыру, қадамдаш орындау. Лог жүргізу және мониторинг. Профильдеу құралдары (CPU, жедел жад). Жадтың ағуын анықтау. IDE ішіндегі ендірілген жөндеу құралдары.

4.1.4. Бағдарламалық жобаларды құрастыруды автоматтандыру құралдары

Бағдарламалық жобаны құрастыру ұғымы. Құрастыру скрипттері мен үдерістері. C/C++ үшін Make, CMake. Java үшін Maven және Gradle. JavaScript жобалары үшін NPM/Yarn. Құрастыруларды контейнерлеу. Тәуелділіктерді басқару. Тестілеу түрлері (unit, integration, UI, жүктемелік тестілеу). Автоматтандырылған тестілеу фреймворктары (JUnit, PyTest, Selenium). Тест сценарийлері және тестімен қамту. Mocking және тәуелділіктерді тестілеу. CI-де тесттерді біріктіру. Тест нәтижелерін есептеу және талдау құралдары.

4.1.5. Бағдарламалық өнімдерді құжаттандыру

Құжаттаманың түрлері (техникалық, пайдаланушыға арналған, API құжаттамасы). Құжаттаманы генерациялау құралдары (Sphinx, Javadoc, Swagger). Құжаттаманың өзектілігін сақтау. Құжаттаманы автоматтандыру. Форматтар: Markdown, HTML, XML, PDF. CASE-құралдар және бағдарламалық жүйелерді модельдеу. CASE-жүйелер ұғымы. UML-диаграммалар (класс диаграммасы, реттілік диаграммасы, пайдаланушы сценарийлері). Деректер қорын жобалауға арналған ER-диаграммалар. Кодты автоматты генерациялау. Rational Rose, Enterprise Architect, Draw.io. Архитектуралық модельдеу.

4.1.6. DevOps құралдары және CI/CD

DevOps тұжырымдамасы және ынтымақтастық мәдениеті. CI/CD құралдары (Jenkins, GitLab CI, GitHub Actions). Контеинеризация: Docker, образдар, Dockerfile. Оркестрация: Kubernetes (шолу). Қолданбаларды автоматты түрде орналастыру. Мониторинг (Prometheus, Grafana). Код сапасын талдау және бақылау құралдары. Кодтың статикалық талдауы. Линтерлер

(ESLint, Pylint, Checkstyle). Осалдықтарды табу құралдары (SonarQube). Code Review және кодтау стандарттары. Бағдарламалық жасақтама сапасының метрикалары. Сапаны тексеруді автоматтандыру.

4.2. Бейіндік пән - «КОМПЬЮТЕРЛІК МОДЕЛЬДЕУ НЕГІЗДЕРІ»

4.2.1. Компьютерлік модельдеуге кіріспе

Модель және модельдеу ұғымы. Модельдер класификациясы: математикалық, имитациялық, статистикалық. Модельді әзірлеу кезеңдері. Python тілінің ғылыми есептеулер мен модельдеудегі рөлі. Модельдеуге арналған Python негіздері: айнымалылар, деректер типтері. Шартты операторлар мен циклдер. Есептеулерге арналған функциялар мен модульдер. Файлдармен жұмыс (модельге арналған деректер).

4.2.2. Python тіліндегі математикалық модельдеу

Линейлік және бейсзық функциялар. Тендеулер жүйелері және оларды шешу (SymPy). Функция графтерін салу (Matplotlib). Сандық әдістер: жұықталған шешімдер. Компьютерлік модельдеудегі сандық әдістер. Эйлер әдісі. Рунге–Кутта әдісі. Жұықталған интегралдау. Дифференциалдық тендеулерді сандық шешу (SciPy).

4.2.3. Статистикалық модельдеу

Кездесоқ шамаларды генерациялау (NumPy). Ікшімалдық үлестірімдері. Монте–Карло модельдеуі. Статистикалық деректерді талдау. Имитациялық модельдеу. Дискреттік оқиғалар. Кезек модельдері. SimPy кітапханасы арқылы процестерді симуляциялау. Уақыттық диаграммаларды құру. Физикалық процестерді модельдеу. Қозғалыс модельдері: жылдамдық, үдеу, траектория. Тербелмелі жүйелер (математикалық маятник). Жылу алмасу модельдері. ОПТ (уақыт бойынша туындысы бар тендеулер, PDE) сандық шешу.

4.2.4. Қеңістіктік модельдеу және визуализация

2D/3D модельдерге арналған деректер генерациясы. Matplotlib 3D, Plotly визуализация құралдары. Процестер анимациясы (FuncAnimation). Беттер мен өрістерді модельдеу. Оптимизациялық модельдер. Сызықтық оптимизация (SciPy.optimize). Функция минимумын іздеу. Оптимизация процестерін модельдеу. Оптимизацияны экономикада, логистикада және IT саласында қолдану.

4.2.5. Модельдеу құралы ретінде машиналық оқыту

Регрессия тәуелділікті модельдеу әдісі. Классификация және шешім қабылдау тәсілдері. Болжамдық модельдерді құру (sklearn). Модельдің дәлдігін бағалау. Құрделі жүйелерді модельдеу. Көпагентті модельдеу. Параллель процестер. Python фреймворктары: Mesa. Жүйелердің мінез-құлқын талдау. Өз моделькі жобалау: мәселені қою; деректерді жинау; модельді Jupyter Notebook-та әзірлеу; нәтижелерді валидациялау және визуализациялау.

4.3. Негізгі пән - «ОБЪЕКТИЛІ-БАҒДАРЛАНГАН ПРОГРАММАЛАУ»

4.3.1. ООП-қа кіріспе

Объект және класс ұғымы. ООП-тың тарихы және маңсаты. Негізгі принциптер: инкапсуляция, мұрагерлік, полиморфизм, абстракция. Бағдарламалық қамтамасыз етуді әзірлеудегі ООП артықшылықтары.

4.3.2. Python тіліндегі кластар мен объектілер

Класс құру (class). Экземплярлар (объектілер). Класс және объект атрибуттары. `__init__()` әдісі (конструктор). Нақты объектілерді модельдеу мысалдары.

4.3.3. Инкапсуляция және қолжетімділік әдістері

Публичті, қорғалған және private атриуттар. Геттерлер және сеттерлер. @property декораторы бар қасиеттер. Деректерді жасыру және қолжетімділікті басқару.

4.3.4. Мұрагерлік

Базалық және туынды кластар. Методтарды қайта анықтау (override). super() арқылы функционалдылықты кеңейту. Көптік мұрагерлік. Полиморфизм. Интерфейс ұғымы. Бірдей интерфейс — әртүрлі мінез-құлық. Мысалдар: жануарлар, көлік, банк операциялары.

4.3.5. Абстракция

Абстрактілі кластар. abc модулі және ABCMeta. Абстрактілі әдістер. Бағдарламалық архитектурада абстракцияны қолдану. Магиялық (dunder) әдістер: `__str__`, `__repr__`. Арифметикалық әдістер: `__add__`, `__sub__`. Салыстыру әдістері: `__eq__`, `__lt__`. Итерация әдістері: `__iter__`, `__next__`. Объект логикасын модельдеуде қолдану.

4.3.6. Коллекциялар және ООП

Пайдаланушы контейнерлері. Итераторлар және генераторлар. Өз құрылымдары: стек, кезек, тізім. Ерекше жағдайлар (exceptions) және қателерді өндөу. try / except. Өз ерекшелік класын құру. Объектілердегі ерекшеліктер. Қателерді логтау.

4.3.7. Композиция және агрегация

«Бөлшек–тұтас» қатынасы. Мұрагерліктен айырмашылығы. Құрделі модельдер құру: автомобиль, университет, брондау жүйесі. Жобалау паттерндері. Файлдармен және объектілермен жұмыс. Графикалық интерфейстер және ООП. Объектілі-бағытталған модельдеу.

4 Тапсырмалар

1. CSV + Pandas + графиктер бойынша сату деректерін талдау
CSV файлын жүктеңіз (мысалы: сату, студенттер, транзакциялар).

Орындаңыз:

- алғашқы жолдарды шығару;
- деректер типтері;
- жетіспейтін мәндер;
- ең көп сатылған 5 санат;
- сатуды айлар бойынша график (line plot) түрінде салу.

2. Статистикалық талдау (NumPy + визуализация)

1000 кездейсоқ саннын тұратын массив генерациялау.

Есептеніз:

- орташа мән, медиана, дисперсия, стандарттық ауытқу;
- бөлу гистограммасын салу;
- Зσ-тен жоғары мәндерді (шығындыларды) анықтау.

3. Мәтіндерді талдау (NLP негіздері + Counter)

Мәтіндік файлды жүктеңіз (немесе интернеттенн мәтін алыңыз).

Орындаңыз:

- тыныс белгілерінен тазарту;
- сөздерді санау;
- сөздердің жиілік талдауы (Counter);
- ең жиі қолданылған 20 сөздің бағаналық диаграммасын салу.

4. GitHub API-мен жұмыс (requests)

GitHub пайдаланушысы туралы деректерді алыңыз:

- логин, жазылуыштар саны, ашық репозиторийлер саны;
- тіркелген күні;

- репозиторий атауларының тізімі;
- жүлдөз саны бойынша ең танымал 5 репозиторийді шығару.

5. Фильмдер деректерін талдау (pandas + топтастыру)
IMDB немесе TMDb ашық датасетін пайдалану.

Орындаңыз:

- рейтинг бойынша үздік 10 фильм;
- жанрлар бойынша орташа рейтинг;
- рейтинг пен бюджет арасындағы тәуелділік графигі;
- өзін-өзі актауы 200%-дан жоғары фильмдерді табу.

6. 3D деректерді визуализациялау (Matplotlib 3D)

3D нұктелерді генерациялау:

- x, y — кездейсоқ;
- $z = \sin(x) + \cos(y)$;
- 3D бет (surface) немесе scatter plot салу.

7. Сызықтық регрессия моделі (sklearn)

Деректер генерациялау:

- x — кездейсоқ;
- $y = 3x + 7 + \text{noise}$.

Орындаңыз:

- LinearRegression моделін оқыту;
- регрессия сызығын визуализациялау;
- MAE, MSE, R² есептеу.

8. Күрделі деңгейлі JSON файлына талдау жасау

JSON жүктеніз (мысалы, ауа райы деректері, GitHub API немесе өз файлыңыз).

Орындаңыз:

- 1-деңгейдегі барлық кілттерді шығару;
- тізім (list) типіндегі элементтерді табу;
- қажетті өрістерді шығару (температура, дата, локация);
- DataFrame-ге түрлендіру.

9. Уақыттық қатарларды талдау (pandas + resample)

Уақытпен байланысты датасет пайдалану (криптовалюта бағасы, айырбас курсы).

Орындаңыз:

- datetime индекске түрлендіру;
- күн/ай бойынша агрегация;
- жылжымалы орташа мән (rolling mean);
- тренд графигін салу.

10. Имитациялық модельдеу (Монте-Карло әдісі)

10 000 симуляция жүргізіңіз:

- акция бағасын кездейсоқ жүріс модельмен модельдеу;
- финалдық мәндердің бөлінісін салу;
- 50 кездейсоқ траекторияны визуализациялау;
- берілген деңгейден төмен түсіп кету ықтималдығын есептеу.

11. Класс және объект құру

Student класы: аты, тобы, орташа балы.

Студент туралы толық ақпаратты шыгаратын әдіс жазыңыз.

12. Инкапсуляция және getter/setter әдістері

Жеке (private) *balance* өрісі бар BankAccount класын жазыңыз.
deposit() және *withdraw()* әдістерін тексерумен іске асырыңыз.

13. Конструкторларды пайдалану

Book класын жасаңыз:

- бос конструктор;
- атауы және авторы қабылданатын конструктор;
- атауы, авторы және жылы бар конструктор.

14. Мұрагерлік (Наследование)

Transport (базалық), Car, Bike кластарын жасаңыз.
move() әдісін базада анықтап, мұрагерлерде қайта анықтаңыз.

15. Полиморфизм

Animal массивін жасаңыз: Dog, Cat, Cow объектілері.

Әрқайсысында *sound()* әдісін жүзеге асырып, полиморфты түрде шақырыңыз.

16. Абстракттілі класс

Абстракттілі Shape класы және *getArea()* әдісі.

Circle, Rectangle кластарын жүзеге асырыңыз.

17. Интерфейстер

ILogger интерфейсін жасаңыз (*log(message)*).

ConsoleLogger, FileLogger — әртүрлі логтау тәсілін іске асырыңыз.

18. Коллекциялармен жұмыс істейтін класс

Library класы:

- кітап қосу;
- жою;
- атауы бойынша іздеу.

19. Ерекше жағдай (Exception) құру

InvalidAgeException ерекше жағдайын жазыңыз.

0–120 диапазонынан тыс жас енгізілсе, қате лақтырысын.

20. Онлайн дүкеннің қарапайым жүйесін жасау

Кластар:

- Product
- Customer
- Order
- OrderItem
- Реализациялау:
 - тауарды тапсырысқа қосу;
 - жалпы құнын есептеу;
 - тапсырыс туралы ақпаратты шығару.

5 Ұсынылатын әдебиеттер

Негізгі әдебиеттер

- 5.1. Гамильтон Р. *Visual Studio 2022: Полное руководство разработчика.* — М.: Вильямс, 2022.
- 5.2. Лутц М. *Изучаем Python.* 5-е изд. — СПб.: Питер, 2021.
- 5.3. Съерра К., Бейтс Б. *Head First: Программирование.* — O'Reilly Media, 2020.
- 5.4. Фаулер М. *Рефакторинг. Улучшение существующего кода.* — СПб.: Питер, 2020.
- 5.5. Буч Г. *Объектно-ориентированный анализ и проектирование с примерами приложений.* — М.: Вильямс, 2019.
- 5.6. Лутц М. *Программирование на Python.* — СПб.: Питер, 2022.
- 5.7. Шилдт Г. *Java. Полное руководство.* — М.: Вильямс, 2021.
- 5.8. Страуструп Б. *Язык программирования C++.* — М.: БХВ-Петербург, 2020.
- 5.9. Гамма Э., Хелм Р., Джонсон Р., Влиссидес Дж. *Паттерны проектирования.* — М.: Вильямс, 2021.
- 5.10. Буч Г. *ООП: принципы и практика.* — М.: Бином, 2020.
- 5.11. Федосеев П. *Компьютерное моделирование: учебник.* — М.: Юрайт, 2021.
- 5.12. Самарский А.А., Михайлов А.П. *Математическое моделирование.* — М.: Наука, 2020.
- 5.13. Higham D., Higham N. *Mathematical Modelling Using Python.* — SIAM, 2019.
- 5.14. Марон И. *Численные методы.* — М.: Физматлит, 2022.
- 5.15. Златопольский Д. *MATLAB и моделирование сложных систем.* — М.: ДМК, 2021.

Қосымша әдебиеттер

- 5.1. McConnell S. *Code Complete.* 2nd Ed. - Microsoft Press, 2019.
- 5.2. Sommerville I. *Software Engineering.* 10th Ed. - Pearson, 2020.
- 5.3. Мартин Р. *Чистая архитектура.* — М.: Питер, 2020.
- 5.4. Канер С., Фолк Дж. *Тестирование программного обеспечения.* — СПб.: Питер, 2021.
- 5.5. Martelli A., Raven W. *Python Cookbook.* -O'Reilly, 2019.
- 5.6. Eckel B. *Thinking in Java.* 4th Ed. - Pearson, 2019.
- 5.7. Bloch J. *Effective Java.* 3rd Ed. - Addison-Wesley, 2020.
- 5.8. Grady Booch. *Object-Oriented Design with Applications.* — Addison-Wesley, 2021.
- 5.9. Liang Y. *Introduction to Java Programming and Data Structures.* — Pearson, 2020.
- 5.10. Muller J. *Computational Modeling and Simulation in Engineering.* — CRC Press, 2019.
- 5.11. Griva I., Nash S. *Linear and Nonlinear Optimization.* — SIAM, 2020.
- 5.12. Guttag J. *Introduction to Computation and Programming Using Python.* — MIT Press, 2021.
- 5.13. Burden R., Faires D. *Numerical Analysis.* — Cengage Learning, 2020.
- 5.14. Allen M. *Python for Computational Science and Engineering.* — Apress, 2021.