

ПРАКТИКАЛЫҚ ЖҰМЫС №6-7

Тақырыбы: Класстар

Мақсаты: C++ тілінде класстарды енгізу бойынша практикалық дағдыларды алу

Қысқаша теориялық мәліметтер

Класс – бұл бірдей сипаттамалары (қасиеттері) бар және бірдей әрекеттер жиынтығын (әдістер, функциялар, процедуралар) орындауға қабілетті нақты дүние объектілерінің жиынтығының абстракциясы. Басқаша айтқанда, класс – нақты дүниедегі бір типті объектілер жиынының әдістері мен қасиеттерінің жиынтығы.

Класс сонымен қатар процедуралар немесе функциялар түрінде олармен жұмыс істеудің ішкі деректері мен әдістері бар пайдаланушы анықтайтын деректер түрі (төменде қараңыз) ретінде түсініледі.

Жалпы алғанда, класс келесідей сипатталады:

```
class имя_класса{  
    модификатор_доступа1:  
    имя_типа1 имя_поля1;  
    ...  
    имя_типаN имя_поляN;  
    модификатор_доступа2:  
    имя_типа1 имя_поляN+1;  
    ...  
    имя_типаN имя_поляN+M;  
    модификатор_доступа3:  
    тип_метода1 имя_метода 1(список_аргументов1);  
    ...  
    тип_метода K имя_метода K(список_аргументовK);  
    модификатор_доступаL:  
    тип_метода K+1 имя_метода K+1(список_аргументовK+1);  
    ...  
    тип_метода K+N имя_метода K+N(список_аргументовK+N);  
}
```

Қол жеткізу модификаторлары үш түрлі болуы мүмкін:

private, **protected** және **public**.

1. **private** (жабық, кластың ішкі мүшесі) - класс мүшесіне кіру тек осы мүше анықталған класс әдістерінен рұқсат етіледі. Класстың кез келген ұрпақтары енді бұл мүшеге кіре алмайды. Жеке түрі бойынша мұраға алу еншілес класстан ата-аналық класстың барлық мүшелеріне, соның ішінде жалпыға ортақ мүшелерге де рұқсат бермейді.

2. **protected** (қорғалған, класс иерархиясының ішкі мүшесі) - осы мүше анықталған класстың әдістерінен, сондай-ақ оның ұрпақ кластарының кез келген әдістерінен класс мүшесіне кіру рұқсат етіледі. Қорғалған мұра негізгі класстың барлық жалпы мүшелерін еншілес класстың қорғалған мүшелері етеді.

3. **public** (класстың ашық түрі) - мүшеге кіруге кез келген кодтан рұқсат етіледі. жалпыға ортақ мұра ата-аналық класс модификаторларын өзгертпейді.

Класстыр идеясы ОББ негізі болып табылады және нақты әлемдегі объектілердің құрылымын көрсетеді, өйткені әрбір объект немесе процесте сипаттамалар жиынтығы немесе ерекше белгілер, басқаша айтқанда, қасиеттер мен мінез-құлықтар бар. Класстар мұрагерлік принциптері бойынша иерархияны құрайды.

Мысалы:

```
#include <iostream>
class Summirovanie {
private:
double a, b;
public:
void seta(double aa){a=aa;}
void setb(double bb){b=bb;}
double sum(){ return a + b; }
};
void main(){
Summirovanie res; //Определим структурную переменную
res. seta(5); res. setb(8); //Зададим значения переменных в res
double c = res. sum(); //Вызовем функцию sum,
//которая вернет значение 13
}
```

Объект немесе нысан (класс данасы) виртуалды кеңістіктегі белгілі бір күйі мен мінез-құлық бар, қасиеттердің (атрибууттардың) мәндерін және олармен операцияларды (әдістерді) берген қандай да бір нысан болып табылады. Нысандар бір немесе бірнеше кластарға жатады, олар өз кезегінде объект үшін қолжетімді болатын қасиеттер мен әдістерді сипаттайды.

Объектілер нақты дүниенің нақты нысандарын көрсету (модельдеу) үшін пайдаланылады. Мысалы, объект келесі атрибуттар жиынтығымен сіздің кір жуғыш машинаңыз болуы мүмкін: «Вятка» өндірістік компаниясы, атауы «Вятка-автомат», өнімнің сериялық нөмірі VYAT454647, сыйымдылығы 20 литр. Объектілер хабарламаларды жіберу және қабылдау арқылы бір-бірімен әрекеттеседі.

Хабарлама – қажетті параметрлер жиынын қамтитын әрекетті орындауға сұраныс. Хабарлама механизмі сәйкес функцияларды немесе класс әдістерін шақыру арқылы жүзеге асырылады.

Кез келген операцияны (әдісті) классқа қосқанда оның қолтаңбасы көрсетілуі керек.

Әдіс қолтаңбасы әдіс атауының, параметр ретінде әдіске жіберілген нысандардың және қайтару түрінің тіркесімі болып табылады.

Интерфейс (спецификация) - объекте анықталған барлық операцияларға арналған қолтаңбалар жиынтығы. Интерфейс нысанға жіберуге болатын барлық сұраулар жинағын сипаттайды.

Деректер түрі - белгілі бір интерфейске (класс атауы) сілтеме жасау үшін қолданылатын атау. Нысан String деп аталатын интерфейсте анықталған операциялардың кез келгенін орындауға сұрауларды қабылдауға дайын болса, String деректер типі деп аталады (бұл тип мысал ретінде алынған).

Конструкторлар (деструкторлар) – класс экземплярларын құру (жою) кезінде автоматты түрде шақырылатын және орындалатын әрекеттер тізбегін орнататын арнайы класс әдістері. Конструктордың аты класстың атымен бірдей, деструктордың аты бірдей, бірақ ~ таңбасының атының алдында қосымшасы бар. Конструкторлар мен деструкторларды кез келген мәнді қайтаратын функциялар ретінде, тіпті жарамсыз деп жариялауға болмайды. Класс аргументтер тізімінде ерекшеленетін бірнеше конструкторларды жүзеге асыра алады, бірақ кез келген класста тек бір деструкторды жүзеге асыруға болады. Деструктордың аргументтері жоқ.

Конструктор/деструктор көмегімен бағдарламаны орындау мысалы

```
#include <iostream>
class INC
{
```

```

private:
int i;
public:
INC(){i=0;} //Конструктор по умолчанию (без аргументов)
INC(int inc){i=inc;} //Конструктор с 1 аргументом
void Inc(){i++;} //Увеличение на 1 значения переменной i
int GetInc(){ //Функция, возвращающая значение переменной i
return i;
}
};
void main(){
INC k; //При создании k вызовется конструктор по умолчанию
INC h(5); //При создании k вызовется конструктор с 1 аргументом
k. Inc(); //В объекте k поле i станет равно 1
k. Inc(); //В объекте k поле i станет равно 2
h. Inc() // В объекте h поле i станет равно 6
}

```

Тапсырма №1 (вариант бойынша)

1. Күрделі сан жұп сандармен (a, b) бейнеленеді, мұндағы a – нақты бөлік, b – елестетілген бөлік. Күрделі сандармен жұмыс істеуге арналған Күрделі класты тәсілдермен орындаңыз: 1) Қосу – қосу: $(a, b) + (c, d) = (a+b, c+d)$; 2) азайту - азайту: $(a, b) - (c, d) = (a-b, c-d)$; 3) Mul – көбейту: $(a,b) \times (c,d) = (ac-bd, ad+bc)$; 4) Div - бөлу: $(a, b) / (c, d) \setminus u003d (ac + bd, bc - ad) / (c^2 + d^2)$.

2. Vector3D класын – үш өлшемді координаттар жүйесіндегі векторды, векторларды қосу және азайту әдістерімен, векторлардың скаляр көбейтіндісін және векторды скалярға көбейтуді жүзеге асырыңыз.

3. Ақшалай сомалармен жұмыс істеу үшін Money класын енгізіңіз. Нөмір екі өрісте көрсетілуі керек: біреуі рубль үшін, екіншісі копейк үшін. Сомаларды қосу, алу, соманы нақты санға бөлу және екі ақшалай соманы салыстыру операциясын орындау қажет.

4. Жазықтықтағы бұрыштармен жұмыс істеу үшін градус және минуттағы мәнмен берілген Angle класын орындаңыз. Операцияларды орындау қажет: бұрыштың мәнін радианға түрлендіру, бұрышты берілген шамаға көбейту және азайту, бұрыш мәнін $0 \div 360$ диапазонына жеткізу.

5. Декарттық координаталар жүйесіндегі жазықтықтағы нүктелермен жұмыс істеу үшін Point класын орындаңыз. Операцияларды жүзеге асыру қажет: берілген мән бойынша x координатасын өзгерту, y координатасын берілген мән бойынша өзгерту, басына дейінгі қашықтықты анықтау, екі нүкте арасындағы қашықтықты есептеу.

6. Рационал (келмейтін) бөлшекті (a, b) жұп сандармен көрсетуге болады, мұндағы a - алым, b - бөлгіш. Рационал бөлшектермен жұмыс істеуге арналған рационал класын әдістермен орындаңыз: 1) Қосу - қосу: $(a, b) + (c, d) = (ad + bc, bd)$; 2) азайту - азайту: $(a, b) - (c, d) = (ad - bc, bd)$; 3) Mul - көбейту: $(a, b) \times (c, d) = (ac, bd)$; 4) Div – бөлу: $(a, b) / (c, d) = (ad, bc)$; 5) Қысқарту – арифметикалық амалдарды орындау кезінде міндетті түрде шақырылатын жеке бөлшекті азайту функциясы.

7. Жыл.ай.күн пішіміндегі күндермен жұмыс істеу үшін Күн классын орындаңыз. Күн үш өрісі бар құрылыммен ұсынылған: жыл, ай және күн. Келесі операцияларды орындау қажет: берілген күндер санынан кейінгі күнді есептеу, күннен берілген күн санын азайту, күндер арасындағы күн санын есептеу, кібісе жылды анықтау.

8. Уақыт классын сағат.минут.секунд пішімінде уақытпен жұмыс істеу үшін енгізіңіз. Уақыт сағат, минут және секунд үш өрісі бар құрылым ретінде ұсынылған. Келесі операцияларды орындау қажет: уақыт пен берілген секунд санын қосу, уақыттан

берілген секунд санын азайту, уақытты секундқа түрлендіру, уақытты минутқа түрлендіру (минутқа дейін дөңгелектеу).

9. Банктік шотпен жұмыс істеу үшін Account класын іске қосыңыз. Класста төрт өріс болуы керек: иесінің тегі, шот нөмірі, есептеу пайызы және рубльдегі сома. Келесі операцияларды орындау қажет: шоттан белгілі бір соманы алу, шотқа белгілі бір соманы салу, пайыздарды есептеу, депозит сомасын долларға айырбастау.

10. Банкоматпен жұмыс істеу үшін Bankomat класын енгізіңіз. Класста келесі өрістер болуы керек: банкоматтың сәйкестендіру нөмірі, банкомат балансы, клиент бір күнде алуға болатын ең төменгі және ең көп сомалар. Келесі операцияларды орындау қажет: банкноттарды банкоматқа салу, белгілі бір ақша сомасын алу (бұл жағдайда алынған соманың дұрыстығын тексеру қажет), банкноттарды алуға болатын максималды соманы өзгерту. клиент бір күнде.

11. Өніммен жұмыс істеу үшін Тауарлар классын енгізіңіз. Класста келесі өрістер болуы керек: өнімнің атауы, тіркеу күні, өнімнің бағасы, өнім бірлігінің саны, шот-фактура нөмірі, оған сәйкес өнім қоймаға жеткізілді. Мынадай операцияларды жүзеге асыру қажет: тауардың бағасын өзгерту, тауардың санын өзгерту (өсу және кему), тауардың өзіндік құнын калькуляциялау.

12. Курсор координаталарымен жұмыс істеу үшін Cursor класын орындаңыз. Класста келесі өрістер болуы керек: курсор координаттары (көлденең және тігінен), курсор түрі - көлденең немесе тік, курсор өлшемі (1-ден 15-ке дейінгі бүтін сан). Операцияларды орындау үшін қажет: курсордың координаталарын өзгерту, курсордың сыртқы түрін өзгерту, курсордың өлшемін өзгерту. Реализовать класс Binary для работы с двоичными числами. В классе должно быть одно поле – число в двоичной форме записи. Требуется реализовать операции: сдвиг влево на заданное количество битов, сдвиг вправо на заданное количество битов, логические операции (and, or, xor, not).

13. Кітаптармен және журналдармен жұмыс істеу үшін Кітаптар классын іске қосыңыз. Класста келесі өрістер болуы керек: кітаптар саны, журналдар саны, басылымдардың максималды саны (кітаптар мен журналдар). Келесі операцияларды орындау қажет: кітаптар санын өзгерту, журналдар санын өзгерту және баспа басылымдарының санын өзгерту.

14. Рюкзактағы заттарды өңдеу үшін Рюкзак классын енгізіңіз. Класста екі өріс болуы керек - заттардың саны және рюкзактың салмағы. Келесі операцияларды орындау қажет: заттардың санын өзгерту, рюкзактың салмағын өзгерту, екі рюкзактарды заттардың саны бойынша салыстыру, екі рюкзактарды салмағы бойынша салыстыру.

15. Үшбұрышты көрсету үшін Triangle класын орындаңыз. Деректер өрістері бұрыштар мен жақтарды қамтуы керек. Есептеу операцияларын орындау қажет: аудан, периметр, сонымен қатар үшбұрыштың түрін анықтау (теңбүйірлі, тең қабырғалы немесе тікбұрышты).

16. Автокөлікпен жұмыс істеу үшін Auto класын жүзеге асыру. Класста төрт өріс болуы керек - газ резервуарының көлемі, газ резервуарындағы бензиннің көлемі, мұнайдың көлемі, автомобильдің жүгірісі. Бензин көлемін, мұнай көлемін және автомобильдің жүрісін өзгерту операцияларын орындау қажет. Әрбір әдісте деректердің дұрыстығын тексеру қажет.

Тапсырма №2

Әрбір класс үшін деструктор мен үш конструкторды енгізу арқылы №1 тапсырманы орындаңыз: аргументтері жоқ, көшіру және инициализациялау.

Бақылау сұрақтары

1. Классты анықтаңыз.
2. Класс қандай элементтерден тұрады?
3. Класс қалай жарияланады?

4. Конструкторды анықтаңыз.
5. Деструкторды анықтаңыз.
6. Конструктор/деструктор қалай жүзеге асырылады?