1. Descrieti pe scurt ce reprezinta obiectul implicit al unei functii.

Obiectul implicit al unei metode este obiectul care apeleaza metoda.

Proprietati:

se apeleaza prin intermediul cuvantului cheie this

modificari produse in functie se reflecta asupra obiectului

nu se mai apeleaza constructor/destructor pentru acesta

2. Descrieti diferenta dintre un pointer si o referinta.

Diferenta dintre un pointer si o referinta este ca un pointer contine adresa spre o zona de memorie in timp ce o referinta este un alias pentru o variabila existenta. In consecinta, o referinta nu poate fi modificata pentru a se referi la alte zone de memorie.

3. Descrieti pe scurt mecanismul de tratare a exceptiilor.

In C++, mecanismul de tratare a exceptiilor este compus din 3 componente: try, throw si catch. Se executa cod in interiorul blocului de try, iar la prima apelare a metodei throw se va iesi din executarea blocului try. Obiectul transmis ca parametru metodei throw este pasat in continuare unui bloc de cod compatibil catch.

4. Descrieti pe scurt diferenta dintre functiile care returneaza valoare si cele care returneaza referinta.

Funcțiile care returnează valoare returnează o copie a datelor, păstrând variabila originală neschimbată. Funcțiile care returnează referință returnează un alias (sau adresă) al variabilei originale, permițând modificarea acesteia direct.

5. Descrieti pe scurt cum se pot defini functii de conversie intre tipuri (clase).

Functiile operator de tip conversie nu trebuie sa aibe parametrii ,trebuie sa fie functii membre, si sa se scrie in antetul lor operator . Singurul parametru pasat este this. Operatorul de tip conversie este de tip unar, argumentul this fiind singurul operand. This indica spre obiectul care va fi supus conversiei. Tipul returnat nu este specificat, deoarece este chiar numele functiei.

Conversia intre tipuri se mai poate realiza si cu ajutorul constructorilor.

Constructorul trebuie sa aibe primul argument diferit de tipul clasei al carei membru este, si in caz ca exista alte argumente, ele trenuie sa fie implicite.O a treia metoda este folosirea operatorului typecast.

6. Spuneti care este diferenta dintre clasa generica (template) si clasa abstracta si ]n ce situatii se foloseste fiecare dintre ele.

O clasa generica poate fi instantiata, si se foloseste atunci cand avem de implementat o clasa ce contine caractesristici generale, si ca algoritmii din program sa foloseasca mai multe tipuri de date. Tipul de date maenvrat in aceasta clasa va fi specificat ca un parametru la crearea obiectelor de tipul acelei clase.

O clasa abstracta nu poate fi instantiate. Ele contin functii virtuale pure, care urmeaza sa fie definite in clasele derivate ce mostenesc o clasa abstracta(ea fiind si clasa de baza). Se foloseste in situatii cand clasa de baza trebuie sa fie generala, elementele specifice fiind atasate in clasele derivate.

7. Descrieti pe scurt diferenta dintre transferul prin valoare si transferul prin referinta al parametrilor n cazul apelului unei functii.

Transferul prin valoare se refera la faptul ca,la apel, valorile transmise vor fi atribuite parametrilor formali. Deci consta in incarcarea parametrilor efectivi in zona de memorie a parametrilor formali(in stiva). Pe cand transmiterea prin referinta se refera la faptul ca, un parametru formal poate fi modificat prin operatiile din corpul functiei, iar la terminarea functiei, in main valorile sunt modificate.

8. Spuneti pe scurt prin ce se caracterizeaza o metoda statica a unei clase.

O metoda statica nu poate avea pointerul this, deoarece ele se pot apela indiferent daca exita un obiect sau nu. Ele pot opera numai asupra variabilelor statice care au fost declarate private. Ele nu pot fi virtuale si nu pot exista doua functii cu acelasi nume, una din ele fiind statica si cealalta non-statica.

9. Spuneti care este diferenta dintre incluziunea de clase si mostenirea de clase si c nd se foloseste fiecare metoda.

Cele 2 metode se folosesc pentru reutilizarea codului. Metoda incluziunii nu este o metoda specifica reutilizareii codului, ea constand includerea unui obiect in altul. In schimb , mostenirea se bazeaza pe un mechanism de ierahizare, clasele derivate mostenind anumiti membrii ai clasei derivate.O alta deosebire ar fi modul de utilizare a claselor existente. La mostenire se doreste utilizarea interfetelor claselor. La includere se utilizeaza proprietatile claselor compuse, si fara sa mosteneasca interfata lor.

10. Descrieti pe scurt constructorul de copiere si situatiile n care acesta este folosit.

Constructorul de copiere este folosit atunci cand se doreste instantierea unui obiect cu altul, apartinand aceleasi clase. Acestia au ca prim argument to refererinta catre clasa ai caror construtor sunt. El are o forma implicita de copiere bit cu bit. Alte utilizari:la transferul unei functii prin valoare, intoarcerea unei valori prin referinta

11. Descrieti diferenta dintre un pointer si o referinta.

Un pointer contine adresa unei zone de memorie si se poate modifica aceasta adresa. Referinta contine o adresa de memorie(locul unde se gaseste aceea variabila) si nu poate fi modificata pentru a referia alte adrese. Spre deosebire de pointeri, referintele nu au operatii special, cum

12. Descrieti pe scurt metoda de identificare a tipului n timpul rularii (RTTI).

RTTI-UL este un mecanism standard care permite identificarea tipului a unui obiect in timpul executiei programului.Uneori dorim sa aflam cel mai derivat obiect cand avem un pointer de tipul unei clase de baza.

Downcastingul se foloseste cand vrem sa castam o referinta a unei clase de bazala una derivata.

Aceasta se realizeaza cu ajutorul functiei typeid din clasa type\_info, si dynamic cast.

Functia typeid intoarce un obiect de tip type\_info. Daca tipul este polimorfic atunci va da informatii despre clasa cea mai derivata.

Functia dynamic\_cast<dest\*>(pointer\_sursa), verifica daca sursa este de acelasi tip ca dest sau daca este baza pentru dest.Se pot compara clase din ierarhii diferite.

13. Spuneti daca o variabila constanta poate fi transmisa ca parametru al unei functii si daca da, n ce situatii. Justificati.

O variabila constanta poate fir transmisa prin parametru, dar parametrul formal nu poate fi schimbat, si nici la intoarcere acea valoare nu poate fi schimbata. Daca se transmite o adresa valoarea nu se poate schimba.

14. Spuneti ce reprezinta o functie virtuala si n ce conditii o functie virtuala defineste o clasa abstracta.

15. Spuneti pe scurt prin ce se caracterizeaza un c mp static al unei clase.

Un membru este static daca la declararea lui este precedat de cuv static .Astfel, I se spune compilatorului ca exista o singura copie a acelei variabile, indifferent de cate instante ale clasei respective exista. Membrul static poate fi accesat indifferent de obiectul appellant.El este instantiat o singura data(la definirea clasei) si este initilizat cu 0, inaintea instantierii oricarui alt obiect. El nu este definit(nu ocupa loc in memorie) in momentul declararii lor. Ele sunt declarate cu operatorul de rezolutie inafara clasei. Ele pot fi apelate cu operatorul de rezolutie cu trimitere la clasa respective.

16. Spuneti ce reprezinta o functie prietena (friend) a unei clase.

O functie friend are accesul la membrii private si protected ai clasei care ii este prietena. Ele pot fi functii ale altor clase. Aceste functii nu sunt membrii clasei respective, deci nu au pointerul this si nu se apeleaza cu ajutorul unui obiect.

17. Descrieti pe scurt ce reguli verifica supra ncarcarea operatorilor.

Se verifica aritatea, daca operatorul este post-fixat sau pre-fixat, si daca este functie membra sau nu. Astfel, se determina numarul dar si tipul de argumente.

18. Descrieti pe scurt mostenirea virtuala si cand poate fi folosita.

In momentul in care sunt doua clase de baza, si ele sunt mostenite de o clasa derivate, pot aparea ambiguitati, deoarece pot exista copii unor membrii al claselor (cum ar fi variabile). Pentru a rezolva astfel de neclaritati, trebuie declarata clasa derivate virtuala. Chiar daca au fost declarate virtual, acei membrii sunt inca prezenti in obiectele de acel tip.

19. Enumerati 3 metode a polimorfismului de compilare

mostenire multipla

mostenire virtuala multipla

compunere; initializarea sub forma de membri a unor alte clase

20. Descrieti pe scurt comportamentul operatorului dynamic\_cast.

Operatorul dynamic\_cast din C++ este utilizat pentru a converti în mod sigur un pointer sau o referință la un tip de clasă bază la un tip de clasă derivată. Acesta funcționează numai cu tipuri de date care conțin polimorfism (adică clase care au cel puțin o funcție virtuală).

21. Descrieti pe scurt functiile virtuale si scopul in care sunt folosite.

Functiile virtuale se folosesc la suprascrierea metodelor in cadrul claselor derivate. Ele sunt accesate printr-o structura de tip pointer. De exemplu, daca avem o metoda cu acelasi nume si in clasa derivate si in clasa de baza, compilatorul va apela functie din clasa de baza. Dar daca ea devine vrituala, ea poate avea alta implementare in fiecara alta clasa derivate. Pnetru a schimba functionalitatea functiei, ea trebuie apelatta printrun pointer catre clasa din cadrul ierarhiei folosite.

22. Descrieti pe scurt cum se comporta destructorii la mosternire.

Destructorii pot fi virtuali, daca dorim sa redefinim un constructor.

In momnentul in care un obiect e pe punctual de a fi distrus, destructorul clasei derivate este primul executat, urmat de destructorul eventualilor membrii ai clasei derivate, ajungandu-se la destructorul clasei de baze.

23. Descrieti pe scurt in ce consta mecanismul de incapsulare.

Mecanismul de incapsulare consta in punerea la un loc a mai multor variabile si metodelor intr-o clasa. Astfel, se poate mentine o mai buna evident a datelor si o gestiune buna a eventualelor erori ce pot aparea.

24. Descrieti pe scurt proprietatile unui camp constant al unei clase.

La definirea membrilor constanti, ele trebui precedate de cuvantul const. Ele vor avea aceeasi valoare pe toata durata existentei obiectului si nu pot fi modificate. Ele pot fi initializate in cadrul constructorului de copiere, prin lista de initializare.

25. Descrieti 3 metode de proiectare diferita prin care prin care elementele unei clase se pot regasi n dublu exemplar, sub diverse forme, n definitia altei clase.

Prima metoda este mostenirea multipla.(functii cu acelasi nume dar tipuri de parametrii diferiti)

A doua este mostenirea ne-virtuala.

A treia este prin compunere.

26. Descrieti pe scurt cum se comporta constructorii la mosternire.

Nu pot fi mosteniti, deci nu pot fi virtuali. Daca constructorul are parametrii, atunci clasa derivata trenuie sa aibe acel constructor. Daca constrcutorul este implicit, atunci clasa de baza poate sa ii lipseasca constructorul. Constructorul clasei de baze este apelat implicit de constructorul clasei derivate.

27. Descrieti pe scurt operatorul typeid.

Operatorul typeid din C++ este utilizat pentru a obține informații despre tipul unui obiect în timpul execuției. Acesta returnează un obiect de tipul std::type\_info, care conține detalii despre tipul obiectului.

28. Descrieti pe scurt functiile constante si dati un exemplu de utilizare.

Functiile constant sunt precedate de cuvantul const in antetul lor. Astfel, se impune ca in cadrul acestor functii sa nu poti modifica valoarea din pointerul this. Nu aveti posibilitatea de a modifica variabilele membru ale obiectului appellant. Daca obiectul este declarat constant ele poate apela doar functii constante. Daca obiectul nu este constant el poate apela functii constante si non-constante.

29. Descrieti pe scurt cele doua feluri de folosire a cuv ntului cheie virtual la mostenire si n ce cazuri se folosesc.

Cuvantul virtual se foloseste la mostenierea virtuala. Atunci cand o clasa derivata mostenest te 2 clase de baza, si in acele clase de baze apar copii a mai multor variabile. Daca dorim sa dorim sa dispara aceasta neclartitate, clasa derivata trebuie sa fie mostenita virtual,si va exista o singura copie.

Cuvantul virtual se foloseste la functii pentru suprascrierea lor in clasele derivate. De exemplu, daca avem o functie in clasa de baza, si alte functii cu acelasi nume in clasele derivate, vor aparea ambiguitati daca vrem sa folosim functia din clasa derivata. Daca le declaram virtual, acesta functie poate avea implentare diferita in mai multe clase derivate.

31. Cum se poate supra ncarca operatorul [] si care este folosirea uzuala a acestei supra ncarcari?

Un exemplu ar fi: char &operator[] (unsigned i). Trebuie sa fie functie membra.Este considerat operat binar. Primul operand va trebuie sa fie un obiect avand acelasi tip cu clasa din care provine. Functia intoarce o referinta.

32. Spuneti pe scurt ce este lista de initializare a unui constructor si subliniati importanta ei.

O lista de inititializare este folosita la initializarea obiectului.Ea contine o lista a menbirlor ce urmeaza a fi initializati si apare in cadrul imprementarii constructorului, intre lista de parametrii si corpul functiei. Ea va fi specificata cu operatorul : . Ea poate fi folosita la initializarea membrilor constanti.

33. Descrieti pe scurt intoarcerea rezultatului unei functii prin referinta.

Întoarcerea rezultatului unei funcții prin referință în C++ permite funcției să returneze un alias la o variabilă existentă, mai degrabă decât o copie a acesteia. Acest lucru poate fi eficient și util în anumite situații.

34. Spuneti pe scurt la ce se refera supra ncarcarea functiilor sablon (template).

Supraincarcarea lor se refera la faptul ca aceste functii pot opera pe mai multe tipuri de date, ele fiind date prin parametru.Astfel putem spune ca functiile template supraincarca automat.Se poate supraincarca o functie generica si explicit, atunci ea va suprascrie template la acea versiune specifica.

Aceste functii nu pot fi virtuale.

35. Descrieti pe scurt diferenta dintre clasa si obiect.

Clasa reuneste metodele si variabilele in cadrul aceluiasi tip de date. Clasa poate ascunde informatia, in vederea protejarii ei.

Obiectul este un tip abstract de dat, entitati, care instantiaza o anumita clasa.

36. Descrieti pe scurt crearea dinamica de obiecte.

Se folosesc opeeratorii new si delete.

Operatorul new este folosit pentru crearea obiectelor.El intoarce un pointer catre zona alocata dynamic, si produce apelul constructorului. De asemenea new stie cata meomorie sa aloce.

Operatorul delete este folosit la distrugerea obiectelor, si este apeleaza destructorul.El elibereaza zona de memorie catre care pointeaza operandul sau.

37. Descrieti pe scurt diferenta dintre polimorfismul de rulare si cel de executie.

Uitate la celalte definitii!

38. Descrieti pe scurt cum se comporta constructorii de copiere la mosternire.

Daca clasa derivata nu are definit un constructor de

copiere atunci compilatorul genereaza unul automat care va

realiza copierea datelor membre mostenite apel nd la

constructorii de copiere ai claselor de baza

, iar datele specifice clasei derivate se vor copia bit cu bit.

Daca clasa derivata are definit un constructor de copiere

atunci acesta se va realiza copierea tuturor datelor membre

(mostenite sau proprii) - apel nd eventual la constructorii

claselor de baza.

39. Descrieti pe scurt in ce consta polimorfismul de executie folosind metode virtuale.

Polimorfismul de executie consta in implementarea diferita a unor functii in clasele derivate.P de exec reperezinta schimbarea dinamica (in timplu rularii) a functionalitatilor definite in urma procesului de derivare.

In momentul in care sunt doua clase de baza, si ele sunt mostenite de o clasa derivate, pot aparea ambiguitati, deoarece pot exista copii unor membrii al claselor (cum ar fi variabile). Pentru a rezolva astfel de neclaritati, este de a declara clasa derivate virtuala. Chiar daca au fost declarate virtual, acei membrii sunt inca prezenti in obiectele de acel tip.

40. Cum se face supraincarcarea ca functii independente in c++.Particularitati.

Supraincarcarea functiilor se realizeaza prin scrierea mai multor functii cu acelasi nume, dar avand numar de parametrii diferiti sau parametrii cu tipuri de date diferite.

Pot aparea amibguitati la conversii, deoarece compilatorul nu stie ce conversii sa faca. De ex ai o functie pt double si float si daca apelezi functia cu un int, va da eroare.Alta problema ar fi daca parametrii au valori implicite sau si cand parametrii sunt transmisi prin valoare sau referinta.

41. Sa se decrie diferenta dintre un pointer constant catre un obiect si un pointer catre un obiect constant.

Unui pointer constant nu i se poate schimba adresa din memorie.

UN pointer constant catre un obiect nu poate face atribuire de adresa.

Un pointer catre un obiect constant poate face atribuire de adresa.

42. Ce este pointerul this?

In momentul in care se apeleaza o functie membru a unei clase, obiectului care a facut apelarea i se va pasa un pointer implicit this.

43. Ce este Polimorfismul?

POLIMORFISMUL reprezinta mecanismul prin intermediul caruia printr-o singura interfata poti manipula o intreaga categorie de actiuni din cadrul unei clase sau a unei ierarhii de clase.

44. Ce este Mostenirea?

MOSTENIREA reprezinta mecanismul prin

care se pot defini noi clase pe baza proprietatilor unor clase deja existente.

45.Descrieti pe scurt ce este si cand se foloseste transferul parametrilor prin referinta constanta