1. האם ניתן לקרוא מתוך פונקציה סטטית של מחלקה לפונקציה של המחלקה שאינה

סטטית? נמק!

תשובה :

לא אפשרי לקרוא לפונקציה לא סטטית מתוך פונקציה סטטית מכיוון שפונקציה סטטית שייכת למחלקה ולא לאובייקט מסויים.

2. האם ניתן להגדיר בתוך מחלקה שתי פונקציות בעלות אותו שם ואותם פרמטרים,

שאחת מהן היא const והשניה לא? למשל, MyClass כוללת את שתי הפונקציות הבאות:

int MyClass::f(int a, int b) const;

int MyClass::f(int a, int b);

תשובה :

ניתן לעשות זאת הקומפיילר ידע לבחור בין הפונקציות לפי האובייקט שנשלח אליהן אם הוא יהיה קונסט או לא .

3. הפונקציה הבאה יוצרת אובייקט ממחלקה ClassA ,ומחזירה מצביע אליו.

ClassA \*func(int var1, int var2)

}

ClassA obj(var1,var2);

return &obj;

{

a .מה לא בסדר בפונקציה זו?

תשובה :

לא נעשתה הקצאה דינאמית בכדי ליצור אובייקט מסוג classA\*

b .כתוב קוד תקין של פונקציה היוצרת אובייקט ממחלקה ClassA ,ומחזירה מצביע אליו.

תשובה :

ClassA \*func(int var1, int var2)

{

ClassA \*obj=new classA(var1,var2);

If(!obj)

{

Cout << "error";

\*obj=NULL;

}

return &obj;

}

4( .6 נקודות( מחלקה MyClass מכילה את השדות הפרטיים הבאים:

class MyClass

}

Private;

int arr\_size;

static int count; //number of objects of this class

float \*arr;

public:

…

;{

כתוב בנאי מעתיק ואופרטור= עבור MyClass.

בנאי מעתיק :

Myclass::Myclass(const Myclass &a)

{

If(a.arr\_size==0)

{

arr\_size = 0;

arr = NULL;

}

Else

{

arr\_size = a.arr\_size;

float \*arr = new float[a.arr\_size];

if(!arr)

{

Cout << "error";

arr = NULL;

{

For (int i = 0 , i<arr\_size,i++;)

{

arr[i] = a.arr[i];

}

Count += arr\_size;

}

}

אופרטור =

Myclass &Myclass::operator=(const Myclass &a)

{

If(a.arr\_size==0)

{

arr\_size = 0;

arr = NULL;

return \*this;

}

Else

{

arr\_size = a.arr\_size;

float \*arr = new float[a.arr\_size];

if(!arr)

{

Cout << "error";

arr = NULL;

return \*this;

{

For (int i = 0 , i<arr\_size,i++;)

{

arr[i] = a.arr[i];

}

Count += arr\_size;

}

return \*this;

}