```
#include <iostream>
using namespace std;

class Base
{
public:
  Base(int newvalue);
  ~Base();
  virtual int foo(int a);
  int count;
private:
  int value;
};
Base::Base(int newvalue) : value(newvalue)
{
  count = 0;
}
Base::~Base()
{}
int Base::foo(int a)
{
  count = count+100; return 10*value + a;
}

class Deriv : public Base
{
public:
  Deriv(int newvalue);
  virtual int foo(int a);
private:
  int value;
  Base* bp;
};
Deriv::Deriv(int newvalue) : Base(newvalue - 2), value(newvalue), bp(new Base(newvalue - 1))
{}
int Deriv::foo(int a)
{
  return 100*value + a;
}

int footwo(Base& b1, Deriv d2)
{
  return b1.foo(1) + d2.foo(2);
}
int main()
{
  Deriv d(9);
  Deriv* dp = new Deriv(5);
  cout << footwo(d, *dp) << endl;
  cout << d.count << endl;
  return 0;
}
```