Programmazione ad oggetti – Laurea in Informatica
Test di meta corso – 11.2.2011

Nome..... Cognome.....

Matricola..... Laurea in.....

Non si possono consultare appunti e libri. Dove previsto scrivere CHIARAMENTE la risposta nell'apposito spazio.

Quesito 1

Si scriva l'output del seguente programma: nel caso si ritenga che il programma non compili correttamente scrivere “NON COMPILA”; nel caso si ritenga che compili correttamente ma si veri chi un errore a tempo di esecuzione scrivere “RUNTIME ERROR” nella parte di output dove si prevede l'errore; scrivere “VALORE CASUALE” quando si prevede che una certa istruzione provochi una stampa di un valore casuale.

```
#include <iostream>
using namespace std;
class C {
public:
  C(int x=0,int y=1) {
     a[0]=x; a[1]=y; cout << "C(" << a[0] << "," << a[1] << ") ";
  }
  int a[2];
};
class D {
public:
  D() : c2(&c1),cr(c1) { cout << "D() ";}
  D(const D& d) : cr(c1) { cout << "Dc ";}
   D() { cout << " D ";}
private:
  C c1;
  C *c2;
  C& cr;
};
class E {
public:
  static C cs;
};
C E::cs;
int main() {
  C c; cout << "UNO" << endl;
  C x(c); cout << x.a[0] << " " << x.a[1] << " DUE" << endl;
  D d=D(); cout << "TRE" << endl;
  E e;cout << "QUATTRO" << endl;
  return 0;
}
```

.....

.....

.....

.....

.....

.....

Quesito 2

Si scriva l'output del seguente programma: nel caso si ritenga che il programma non compili correttamente scrivere “NON COMPILA”; nel caso si ritenga che compili correttamente ma si verifichi un errore a tempo di esecuzione scrivere “RUNTIME ERROR” nella parte di output dove si prevede l'errore; scrivere “VALORE CASUALE” quando si prevede che una certa istruzione provochi una stampa di un valore casuale.

```
#include <iostream>
using namespace std;
class C {
public:
  C(int n=0) : number(n) { cout << "C(" << number << ") "; }
   C() { cout << " C "; }
  C(const C& c) : number(c.number) { cout << "Cc(" << number << ") "; }
  operator int() { cout << "int() "; return 3;}
private:
  int number;
};
int F(C c) {return c;}
int main() {
  C *c=new C; cout << "UNO" << endl;
  C d; cout << "DUE" << endl;
  int x=F(d); cout << "TRE" << endl;
  C e=F(d); cout << "QUATTRO" << endl;

  return 0;
}
```

.....

.....

.....

.....

.....

.....

Quesito 3

Si consideri il seguente frammento di codice:

```
namespace ns {
  class C {
  public:
    C(int n=0) : x(n) {}
  private:
    friend int f();
    int x;
  };
}
int f() {
  ns::C c;
  return c.x;
}
int main() {
  f();
  return 0;
}
```

Barrare con una croce quali tra le seguenti affermazioni è esatta.

1. non compila perché `f` non può accedere alla parte privata di `c`
2. non compila perché `f` è una funzione privata

3. da un errore di accesso illegale a runtime
4. compila, linka ed esegue correttamente

Quesito 4

Si scriva l'output del seguente programma: nel caso si ritenga che il programma non compili correttamente scrivere “NON COMPILA”; nel caso si ritenga che compili correttamente ma si veri chi un errore a tempo di esecuzione scrivere “RUNTIME ERROR” nella parte di output dove si prevede l'errore; scrivere “VALORE CASUALE” quando si prevede che una certa istruzione provochi una stampa di un valore casuale.

```
#include <iostream>
using namespace std;
class C {
public:
  C(int n=1) : number(n) { cout << "C(" << number << ") "; }
   C() { cout << " C(" << number << ") "; }
  C& operator=(const C& c) { number=c.number; cout << "operator=(" << number << ") "; }
  int number;
};

int F(C c) {return c.number;}

int main() {
  C *c=new C; cout << "UNO" << endl;
  C d; d=*c; cout << "DUE" << endl;
  int x=F(d); cout << "TRE" << endl;
  int y=F(F(d)); cout << "QUATTRO" << endl;

  return 0;
}
```

.....

.....

.....

.....

.....

.....

Quesito 5

Si scriva l'output del seguente programma: nel caso si ritenga che il programma non compili correttamente scrivere “NON COMPILA”; nel caso si ritenga che compili correttamente ma si veri chi un errore a tempo di esecuzione scrivere “RUNTIME ERROR” nella parte di output dove si prevede l'errore; scrivere “VALORE CASUALE” quando si prevede che una certa istruzione provochi una stampa di un valore casuale.

```
#include <iostream>
using namespace std;
class C {
public:
  C(int x=0,int y=1) {a[0]=x; a[1]=y; cout << "C(" << a[0] << "," << a[1] << ") ";}
  C(const C&) {cout << "Cc ";}
  int a[2];
};
class D {
public:
  D() : c2(&c1),cr(c1) { cout << "D() ";}
  D(const D& d) : cr(c1) { cout << "Dc ";}
   D() { cout << " D ";}
private:
  C c1;
  C *c2;
  C& cr;
};
class E {
public:
  static C cs;
};
C E::cs=1;

int main() {
  C c; cout << "UNO" << endl;
  C x(c); cout << x.a[0] << " " << x.a[1] << " DUE" << endl;
  D d=D(); cout << "TRE" << endl;
  E e;cout << "QUATTRO" << endl;
  return 0;
}
```

.....

.....

.....

.....

.....

.....