UNIVERSITÀ DELLA CALABRIA
DIPARTIMENTO DI MATEMATICA E INFORMATICA

Corso di Laurea In Informatica
Esame di Programmazione ad Oggetti
07/02/2017
Tempo a Disposizione: 60 min

Cognome: _____ Nome: _____

Matricola: _____ Esercizi lab: _____

```
 1: template<class T>
 2: class Stack{
 3: public:
 4:    Stack(int n):size(0),capacity(n){array=new T[n];};
 5:    Stack();
 6:    Stack(const Stack<T>& s);
 7:    Stack<T>& operator=(const Stack<T>& s);
 8:    ~Stack();
 9:    bool push(T object);
 10:   bool pop();
 11:   unsigned getSize() const;
 12: protected:
 13:      T* array;
 14:      unsigned size;
 15:      unsigned capacity;
};

//Completare opportunamente il main (max 2 punti)
int main() {
    Stack<int> s1(3);
    s1.push(1);
    s1.push(2);

    Stack<string*>* s2 = new Stack<string*>(5);
    string* stringa1 = new string("ciao");
    string* stringa2 = new string("hello");
    string* stringa3 = string1;
    s2->push(stringa1);
    s2->push(stringa2);
    s2->push(stringa3);









    return 0;
}

//Implementare i seguenti metodi (max 8 punti)
template <class T> bool Stack<T>::pop()
{









}
```

Dipartimento di Matematica e Informatica
Università della Calabria
Via P. Bucci, Cubo 30B
87036 Rende (CS) - ITALIA

Tel. (+39) 0984.496402
Fax (+39) 0984.496410
dipartimento.demacs@unical.it
dipartimento.demacs@pec.unical.it

www.mat.unical.it

```
template <class T> bool Stack<T>::push(T object)
{









}
template <class T> Stack<T>::Stack()
{









}
```

```
 1: class Linea{
 2: public:
 3:    Linea(const string& s);
 4:    Linea();
 5:    const string& getLinea() const;
 6:    void setLinea( const string&);
 7:    virtual void stampa() const {cout << linea << endl;};
 8: private:
 9:      string linea;
};

//Definire opportunamente la seguente classe (max 2 punti):
class LineaAllineataADestra: ___________ Linea {
   public:









      virtual void stampa() const;
   private:
      unsigned numeroSpaziADestra;


   protected:


};
```

Dipartimento di Matematica e Informatica
Università della Calabria
Via P. Bucci, Cubo 30B
87036 Rende (CS) - ITALIA

Tel. (+39) 0984.496402
Fax (+39) 0984.496410
dipartimento.demacs@unical.it
dipartimento.demacs@pec.unical.it

www.mat.unical.it

Programmazione Ad Oggetti. 7 Febbraio 2017

Cognome: _____ Nome: _____

Matricola: _____ Esercizi lab: _____

Rispondere alle seguenti domande a risposta multipla (3 punti per risposta e motivazione esatta, -2 punti per risposta sbagliata, 0 per risposta non data o risposta esatta ma priva di motivazione):

1. Quale tra le seguenti implementazioni è corretta?

   a) `template<class T> Stack<T>::~Stack() {delete[] array;}`

   b) `template<class T> Stack<T>::~Stack() {delete array;}`

   c)
   ```
   template<class T> Stack<T>::~Stack() {
         for(int i = 0; i < size; i++)
             delete array[i];
         delete [] array;}
   ```

   c) `template<class T> Stack<T>::~Stack() {}`

   Motivazione:

2. Quale tra le seguenti implementazioni è corretta?

   a) `template<class T> unsigned Stack<T>::getSize() const {return array.size;}`

   b) `template<class T> unsigned Stack<T>::getSize() const {return size();}`

   c) `template<class T> unsigned Stack<T>::getSize() const {return size;}`

   d) `template<class T> unsigned Stack<T>::getSize() const {return capacity;}`

   Motivazione:

3. Quale tra le seguenti implementazioni è corretta?

   a)
   ```
   template<class T> Stack<T>& Stack<T>::operator=(const Stack<T>& s) {
       if(this!=&s){
           delete array;
           size = s.size;
           capacity = s.capacity;
           array = 0;
           if(s.array!=0){
               array = new T[capacity];
               for(unsigned i = 0; i < size; i++)
                   s.array[i] = array[i];
           }
       }
       return this;
   }
   ```

   b)
   ```
   template<class T> Stack<T>& Stack<T>::operator=(const Stack<T>& s) {
       if(this!=&s){
           delete[] array;
           size = s.size;
           capacity = s.capacity;
           array = 0;
           if(s.array!=0){
               array = new T[capacity];
               for(unsigned i = 0; i < size; i++)
                   array[i] = s.array[i];
           }
       }
       return *this;
   }
   ```

c)
```
template<class T> Stack<T>& Stack<T>::operator=(const Stack<T>& s) {
    if(this!=&s){
        ~Stack();
        size = s.size;
        capacity = s.capacity;
        array = 0;
        if(s.array!=0){
            array = new T[capacity];
            for(unsigned i = 0; i < size; i++)
                array[i] = s.array[i];
        }
    }
    return *this;
}
```

d) Nessuna delle precedenti

Motivazione:

---

4. Quale tra le seguenti implementazioni è corretta?

a)
```
virtual void LineaAllineataADestra::stampa() const {
    for(unsigned i=0;i<numeroSpaziADestra;i++)
        cout<<" ";
    cout<<linea<<endl;
}
```

b)
```
virtual void LineaAllineataADestra::stampa() const {
    for(unsigned i=0;i<numeroSpaziADestra;i++)
        cout<<" ";
    stampa();
}
```

c)
```
virtual void LineaAllineataADestra::stampa() const {
    for(unsigned i=0;i<numeroSpaziADestra;i++)
        cout<<" ";
    Linea.stampa();
}
```

d)
```
virtual void LineaAllineataADestra::stampa() const {
    for(unsigned i=0;i<numeroSpaziADestra;i++)
        cout<<" ";
    Linea::stampa();
}
```

Motivazione:

---

5. Sia **class LineaCentrata: private Linea** la definizione di una classe LineaCentrata. Quale tra le seguenti istruzioni **è consentita** nel metodo stampa di LineaCentrata?

a) `cout << getLinea() << endl;`

b) `linea = "";`

c) `Sono entrambe consentite`

d) `Sono entrambe sbagliate`

Motivazione:

---

6. Sia **class LineaCentrata: private Linea** la definizione di una classe LineaCentrata. Quale tra le seguenti istruzioni **è consentita** nel main?

a) `LineaCentrata l; l.getLinea();`

b) `LineaCentrata l; l.linea="";`

c) `Sono entrambe consentite`

d) `Sono entrambe sbagliate`

Motivazione:
