

Nome:
Cognome:
Matricola:

1. Presentate le principali funzioni per la gestione della memoria dinamica in C.

Sol. Si vedano le funzioni **malloc**, **calloc** e **free** (slide e libro di testo).

2. Definite una macro **max(a,b)** da utilizzare per calcolo del massimo tra due variabili **a** e **b**. Esistono dei limiti al suo uso?

Sol. Ricorrendo all'operatore ternario **?:**, possiamo scrivere

```
#define max(a,b) ((a)>(b)?(a):(b))
```

Inevitabilmente, non è possibile evitare incrementi multipli nel caso di operatori con side effect, ad esempio **max(++a,++b)** causa un doppio incremento di una delle due variabili.

3. Definite una procedura da utilizzare per lo scambio dei valori contenuti in due variabili float. In altri termini, dopo l'esecuzione delle istruzioni

```
float a=8.9;
float b=9.8;
Scambio(    );
```

a deve valere 9.8 e **b** 8.9.

Sol. Poichè in C l'unica regola per il passaggio dei parametri è quella per valore (o copia), è necessario che la procedura conosca l'indirizzo delle due variabili di cui scambiare il contenuto.

```
void Scambio(float* a,float*b)
{float tmp;
 tmp=*a; *a=*b; *b=tmp;
}
```

Scambieremo quindi **a** e **b** tramite la chiamata **Scambio(&a,&b)**.

4. Illustrate con esempi i possibili usi della parola riservata **virtual**.

Sol. Si vedano le funzioni virtuali e le classi base virtuali (lucidi e libro di Stroustrup).

5. Completate la seguente dichiarazione

```
class Complex {
double re; double im;
public:
...};
```

in modo tale che, dati tre oggetti **Complex a, b e c**, l'espressione **a=2+b+c** venga correttamente compilata.

Sol. Occorre effettuare l'overloading dell'operatore **+** attraverso la definizione di una funzione amica in grado di sommare due complessi. In aggiunta, deve essere definito almeno un costruttore per un complesso a partire da un intero. Ad esempio,

```
class Complex {
double re; double im;
public:
friend Complex operator+(const complex& a,const complex& b)
{return Complex(a.re+b.re,a.im+b.im);};
Complex(double x=0.0,double y=0.0){re=x;im=y;};
};
```

6. Illustrate l'ordine in cui vengono chiamati i costruttori coinvolti nella creazione di un oggetto di una classe **C** che eredita da una classe **B** la quale eredita da una classe **A**. Supponete che all'interno di **B** ci sia un membro che è un oggetto di una classe **D** e che all'interno di **A** ci sia un membro che è un oggetto di una classe **E**.

Sol. Il costruttore di una classe derivata viene chiamato dopo quello della classe base. Inoltre, se tra i membri di una classe ci sono degli oggetti, i costruttori di questi sono chiamati prima del costruttore della classe. L'ordine è quindi il seguente: costruttore di **E**, costruttore di **A**, costruttore di **D**, costruttore di **B** e infine costruttore di **C**.