

Mereghetti	<input type="checkbox"/>
Spoletini	<input type="checkbox"/>
Tamascelli	<input type="checkbox"/>

INFORMATICA - scritto - 20 settembre 2018

Cognome _____ Nome _____ Matr _____

Lab (voto/quando) _____ / _____ Firma _____

1) Siano date le seguenti definizioni:

```
struct elem {
    int item;
    int *p;
};

elem *X = new elem[100]; elem *z: int y;
```

- Come assegnare al campo **item** dell'elemento di **X** di indice **3** il valore del campo **item** dell'ultimo elemento di **X**?

- Come assegnare l'indirizzo di **y** al campo **p** del primo elemento di **X**?

- Come creare dinamicamente una variabile intera e farla puntare dal campo **p** dell'elemento puntato da **z**?

2) Quale sarà il valore finale delle variabili **a** e **b** dopo l'esecuzione del seguente frammento di codice?

```
int A[7] = {1, 2, 3, 4, 5, 6, 7};
int a = 0; float b = 0;

for ( int i = 0; i < 7; i++ )
    if ( A[i]%2 == 1 )
        a = a + ++i;
a = _____

for ( int i = 0; i < 7; i += 2 )
    b = b + A[i]/2.0;
b = _____
```

3) Inizializzate la variabile **matr** col vostro numero di matricola e dite quanto valgono le variabili **a**, **c** e **x** al termine dell'esecuzione del codice seguente:

```
int matr = _____;
float a = matr;
int c = (matr/1000)*1000 + matr%1000;
float a = (a/1000)*1000 + c;;
float x = (int) matr/2.0 + (int)(matr/2.0)
```

a = _____ c = _____ x = _____

4) La struttura

```
struct cerchio {  
    float x,y; /* coordinate del centro */  
    float r;   /* raggio */  
}
```

rappresenta un cerchio nel piano cartesiano avente centro nel punto di coordinate **(x,y)** e raggio **r**. Scrivete un frammento di codice che utilizzi la struttura **cerchio** per memorizzare 100 cerchi inseriti dall'utente, *impedendo* l'inserimento di cerchi con raggio nullo o negativo. Successivamente, il codice deve stampare i soli cerchi che *giacciono completamente nel primo quadrante*.

5) Scrivete la funzione

```
bool secPari(int *X, int dimX, int p)
```

che accetta in ingresso un array di interi *positivi* **X** di dimensione **dimX** e un intero *positivo* **p**. La funzione deve restituire **true** se **X** contiene una sequenza di almeno **p** numeri pari *consecutivi*, **false** altrimenti. Ad esempio, assumendo l'array **X = {5, 6, 28, 4, 13, 13}**, la chiamata **secPari(X, 6, 2)** deve restituire **true**.