

**FONDAMENTI DI PROGRAMMAZIONE A***Tempo a disposizione: 20 minuti*

Nome ..... Cognome ..... Matricola .....

*Per accedere alla prova di programmazione è necessario rispondere correttamente ad almeno il 70% delle domande*

1. Si consideri il seguente frammento di codice.

```
float var1 = 5;  
float& var2 = var1;
```

Si indichi la risposta corretta

- a var2 contiene l'indirizzo di memoria della variabile var1
- b il frammento di codice ritorna un errore a tempo di compilazione
- c var2 è un riferimento alla variabile var1
- d il frammento di codice compila ma ritorna un errore a tempo di esecuzione
- e nessuna delle precedenti

2. Supponendo che le espressioni vengano valutate da sinistra verso destra, cosa stampa il seguente frammento di codice?

```
int w = 2, z = 2;  
bool b = w == ++z && ++w == --z + w;  
cout << b << " " << w << " " << z << endl;
```

- a 1 2 3
- b 1 3 3
- c 0 2 3
- d 0 2 2
- e nessuna delle precedenti

3. Si consideri la seguente dichiarazione: `char s[] = {'a', 'b', 'c', '\0', 'd'};`. La variabile s corrisponde

- a alla stringa C-style "abc"
- b alla stringa C-style "abcd"
- c alla stringa C-style "" (stringa vuota)
- d non è una stringa C-style

4. Cosa stampa il seguente frammento di codice?

```
int x = 1;  
switch (x) {  
    case 1: cout << "Uno";  
    case 2: cout << "Due";  
    default: cout << "Default"; break;  
}
```

- a Uno
- b UnoDue
- c UnoDueDefault
- d Default
- e nessuna delle precedenti

5. Gli array prevedono un accesso di tipo

- a sequenziale
- b diretto
- c LIFO (*Last In First Out*)
- d FIFO (*First In First Out*)
- e nessuna delle precedenti

6. Sia `ptr` un puntatore a caratteri. L'espressione `ptr++`

- a ritorna un errore a tempo di compilazione
- b incrementa l'indirizzo di memoria contenuto in `ptr`
- c compila ma ritorna un errore a tempo di esecuzione
- d incrementa il valore della variabile puntata da `ptr`
- e nessuna delle precedenti

7. Si indichi cosa viene stampato dalla funzione `main`

```
void foo(int& x) {  
    x--;  
}
```

```
int main() {  
    int v = 3;  
    foo(v);  
    cout << v;  
    return 0;  
}
```

- a 2
- b 3
- c non compila
- d comportamento indefinito
- e nessuna delle precedenti

8. Si indichi quale operatore viene utilizzato per referenziare una variabile

- a .
- b \*
- c ->
- d &
- e nessuna delle precedenti

9. Dato un puntatore a caratteri `ptr`, è possibile eseguire l'operazione `ptr + 3`.

T  F

10. In C++, se una zona di memoria allocata nello heap non è puntata da nessun puntatore, è considerata *garbage* e viene deallocated automaticamente.

T  F