

FONDAMENTI DI PROGRAMMAZIONE A*Tempo a disposizione: 30 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.

```
int v = 5;
int* w = &v;
```

Si indichi la risposta corretta

- ☐ *a* *w* contiene l'indirizzo di memoria della variabile *v*
☐ *b* il frammento di codice ritorna un errore a tempo di compilazione
☐ *c* *w* è un riferimento alla variabile *v*
☐ *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 a monitor il seguente frammento di codice?

```
int y = 1, z = 2;
bool b = ++y == z++ || ++y == --z + y;
cout << b << " " << y << " " << z << endl;
```

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

3. Si consideri la seguente dichiarazione: `char s[] = {'h', '\0'};`. La variabile `s` corrisponde alla stringa C-style "h".

☐ *T* ☐ *F*

4. Si indichi cosa stampa il seguente programma.

```
for (int i = 4; i > 0; i--) {
    if (i <= 3)
        break;
    cout << i << " ";
}
```

- ☐ *a* 4
☐ *b* 4 3 2
☐ *c* 4 3 2 1
☐ *d* 4 3 2 1 0

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. Cosa stampa il seguente programma?

```
int V[] = {5,11,20,17,8};  
int* p = &(V[1]);  
cout << *(p + 2);
```

- ☐ a 11 ☐ b 20 ☐ c 10 ☐ d 17 ☐ e 8

7. Si consideri il seguente tipo struct

```
struct S {  
    int x;  
    int y;  
}
```

e la seguente dichiarazione: `S s = {1, 2};`. L'espressione corretta per accedere al campo `y` di `s` è

- ☐ a `(&s).y`
☐ b `s->y`
☐ c `s[y]`
☐ d `s.y`
☐ e nessuna delle precedenti

8. Per quali valori di `x`, variabile di tipo `int`, l'espressione `x > 0 && x < 10` risulta vera?

- ☐ a nessun valore intero
☐ b qualsiasi valore intero
☐ c esclusivamente i valori compresi fra 1 (incluso) e 9 (incluso)
☐ d esclusivamente i valori minori strettamente di 10
☐ e nessuna delle precedenti

9. Si indichi cosa stampa il seguente frammento di codice

```
int x = 3, y = 2;  
int* p1 = &x;  
int* p2 = &y;  
*p2 = *p1 + 1;  
cout << x << "□" << y << endl;
```

- ☐ a 3 4 ☐ b 3 2 ☐ c 4 4 ☐ d 4 3 ☐ e nessuna delle precedenti

10. In C++, il passaggio di parametri per riferimento è implementato esclusivamente tramite puntatori.

☐ T ☐ F