

Corso: Fondamenti di Informatica 1 (gruppo 8-9)

Corsi di laurea: Area dell'Informazione

Questionario di autovalutazione 4

Domanda n. 1

Argomento: Rappresentazione dell'informazione

Peso: 1

Pubblica: N

Domanda: Il numero binario a 8 bit in complemento a due 0100 1111 corrisponde al numero decimale

Risposte:

1. -78

Punteggio: 0

2. 74

Punteggio: 0

3. 84

Punteggio: 0

4. 138

Punteggio: 0

5. 79

Punteggio: 1

Spiegazione:

0100 1111 in base 2 = $2^6 + 2^3 + 2^2 + 2^1 + 2^0 = 64 + 8 + 4 + 2 + 1 = 79$ in base 10

Domanda n. 2

Argomento: Rappresentazione dell'informazione

Peso: 1

Pubblica: N

Domanda: Il numero binario a 8 bit in complemento a due 1111 1100 corrisponde al numero decimale

Risposte:

1. -3

Punteggio: 0

2. 252

Punteggio: 0

3. -4

Punteggio: 1

4. -124

Punteggio: 0

Spiegazione:

Applicando la definizione di complemento a due a 8 bit: 1111 1100 in base 2 = $-2^7 + 2^6 + 2^5 + 2^4 + 2^3 + 2^2 = -128 + 64 + 32 + 16 + 8 + 4 = -4$ in base 10

In alternativa si calcoli l'inverso del numero: 0000 0011 + 1 = 0000 0100 in base due = 4 in base dieci.

Domanda n. 3

Argomento: Rappresentazione dell'informazione

Peso: 1

Pubblica: N

Domanda: Quale e' la rappresentazione binaria (a 8 bit, in complemento a due) del numero -37 ?

Risposte:

1. 0101 1101

Punteggio: 0

2. 0010 1011

Punteggio: 0

3. 0010 0011

Punteggio: 0

4. 1101 1011

Punteggio: 1

5. il numero dato non puo' essere rappresentato in complemento a due a 8 bit

Punteggio: 0

Spiegazione:

Si calcoli la rappresentazione del numero decimale +37 e poi si applichi l'algoritmo per l'inversione del numero.

37 in base dieci = $32 + 4 + 1 = 0010\ 0101$ in base 2; -37 in base dieci = $1101\ 1010 + 1 = 1101\ 1011$ in base 2.

Domanda n. 4

Argomento: Rappresentazione dell'informazione

Peso: 1

Pubblica: N

Domanda: Cambiando segno al numero binario in complemento a due 1111 1110 si ottiene:

Risposte:

1. 0000 0010

Punteggio: 1

2. 0000 0001

Punteggio: 0

3. 0111 1110

Punteggio: 0

4. Viene generata una condizione di overflow (trabocco)

Punteggio: 0

5. Nessuna delle precedenti

Punteggio: 0

Spiegazione:

Si applichi l'algoritmo di inversione (complemento a 2).

Domanda n. 5

Argomento: Variabili e tipi

Peso: 1

Pubblica: N

Domanda: In Java il tipo di dati fondamentali char:

Risposte:

1. e' un numero intero compreso fra 0 e 127 estremi inclusi

Punteggio: 0

2. e' un numero intero compreso fra 0 e 1023 estremi inclusi

Punteggio: 0

3. e' un numero intero compreso fra 0 e 65535 estremi inclusi

Punteggio: 1

4. non e' un numero intero

Punteggio: 0

5. nessuna delle precedenti affermazioni e' corretta

Punteggio: 0

Spiegazione:

Il tipo di dati fondamentali char in Java e' un numero intero positivo espresso a 16 bit.

Domanda n. 6

Argomento: Iterazioni, array e stringhe

Peso: 1

Pubblica: N

Domanda: Il seguente frammento di codice in Java:

```
final int A = 0;  
int b = A++;  
System.out.println(b);
```

Risposte:

1. compila e in esecuzione invia a standard output il numero 1

Punteggio: 0

2. compila ma in esecuzione genera un'eccezione

Punteggio: 0

3. compila e in esecuzione invia a standard output il numero 0

Punteggio: 0

4. genera un errore di compilazione

Punteggio: 1

5. nessuna delle precedenti affermazioni e' corretta

Punteggio: 0

Spiegazione:

La variabile A e' una costante, quindi non puo' essere modificata nell'enunciato A++.

Domanda n. 7

Argomento: Variabili e tipi

Peso: 1

Pubblica: N

Domanda: Il seguente frammento di codice

```
public class X {  
    ...  
    private double y;  
}
```

Risposte:

1. rende la variabile di esempio y accessibile da qualsiasi metodo di qualsiasi classe

Punteggio: 0

2. rende la variabile di esempio y accessibile da tutte le classi che usano esemplari della classe X

Punteggio: 0

3. rende la variabile di esempio y accessibile solo dai metodi della classe X

Punteggio: 1

4. rende costante la variabile di esempio y

Punteggio: 0

5. nessuna delle precedenti affermazioni e' corretta

Punteggio: 0

Spiegazione:

Le variabili di esempio private sono visibili solo nella classe in cui sono definite.

Domanda n. 8

Argomento: Variabili e tipi

Peso: 1

Pubblica: N

Domanda: Qual e' il significato della parola chiave this in un metodo di esempio in Java

Risposte:

1. il metodo stesso

Punteggio: 0

2. la classe a cui appartiene il metodo

Punteggio: 0

3. il parametro implicito della chiamata al metodo

Punteggio: 1

4. il costruttore predefinito della classe

Punteggio: 0

5. nessuna delle precedenti affermazioni e' corretta

Punteggio: 0

Spiegazione:

Indica il parametro implicito della chiamata al metodo.

Domanda n. 9

Argomento: Iterazioni, array e stringhe

Peso: 1

Pubblica: N

Domanda: Il metodo seguente

```
public static int method(String s)
{ return 2 * s.length();
}
```

Risposte:

1. genera uno o piu' errori in compilazione

Punteggio: 0

2. lancia un'eccezione se riceve come parametro il valore null

Punteggio: 1

3. lancia un'eccezione se riceve come parametro la stringa vuota

Punteggio: 0

4. non lancia mai eccezioni

Punteggio: 0

5. nessuna delle affermazioni precedenti e' corretta

Punteggio: 0

Spiegazione:

L'invocazione di un metodo non statico con parametro implicito pari a null provoca sempre l'eccezione NullPointerException.

Nel caso specifico, s.length() lancia l'eccezione NullPointerException se s = null.

Domanda n. 10

Argomento: Logica booleana

Peso: 1

Pubblica: N

Domanda: Quale delle espressioni logiche in Java sotto riportate e' equivalente alla seguente?

```
!(a >= 0 || a <= 10)
```

Risposte:

1. $a < 0 \parallel a > 10$

Punteggio: 0

2. $a \leq 0 \&\& a \geq 10$

Punteggio: 0

3. $a \geq 0 \parallel a \leq 10$

Punteggio: 0

4. $a > 0 \&\& a < 10$

Punteggio: 0

5. nessuna delle precedenti risposte e' corretta

Punteggio: 1

Spiegazione:

Si applichino le leggi di De Morgan:

`!(a >= 0 || a <= 10)` equivale a `(a < 0 && a > 10)`

Domanda n. 11

Argomento: Logica booleana

Peso: 1

Pubblica: N

Domanda: Negando la seguente espressione logica in Java mediante la legge di De Morgan si ottiene:

`(a || !b) && (!a && b)`

Risposte:

1. `(a || !b) || (!a && b)`

Punteggio: 0

2. `(a && !b) || (!a || b)`

Punteggio: 0

3. `(!a && b) || (a || !b)`

Punteggio: 1

4. `(!a && b) && (a || !b)`

Punteggio: 0

5. nessuna delle precedenti risposte e' corretta

Punteggio: 0

Spiegazione:

Si applichino le leggi di De Morgan:

`!((a || !b) && (!a && b))` equivale a `(!a && b) || (a || !b)`

Domanda n. 12

Argomento: Logica booleana

Peso: 1

Pubblica: N

Domanda: Quale delle espressioni logiche sotto riportate e' equivalente alla seguente?

`!((a > 0) && (a < 10))`

Risposte:

1. `a <= 0 && a >= 10`

Punteggio: 0

2. `a < 0 || a > 10`

Punteggio: 0

3. `a <= 0 || a >= 10`

Punteggio: 1

4. `a < 0 && a > 10`

Punteggio: 0

5. nessuna delle precedenti risposte e' corretta

Punteggio: 0

Spiegazione:

Si applichino le leggi di De Morgan:

`!((a > 0) && (a < 10))` equivale a `a <= 0 || a >= 10`

Domanda n. 13

Argomento: Logica booleana

Peso: 1

Pubblica: N

Domanda: Con riguardo alla seguente espressione logica quali affermazioni sono vere:

$((a \ \&\& \ b) \ || \ (!a \ \&\& \ !b)) \ == \ !((a \ || \ b) \ \&\& \ (!a \ || \ !b))$

Risposte:

1. e' sempre verificata, indipendentemente dai valori da a e b

Punteggio: 1

2. la relazione e' sempre verificata perche' le espressioni a destra e a sinistra dell'uguale valgono entrambe true per qualsiasi valore delle variabili a e b

Punteggio: 0

3. la relazione e' sempre verificata perche' le espressioni a destra e a sinistra dell'uguale valgono entrambe false per qualsiasi valore delle variabili a e b

Punteggio: 0

4. la relazione non e' mai verificata perche' l'espressione a sinistra vale true mentre quella a destra vale false per qualsiasi valore delle variabili a e b

Punteggio: 0

5. la relazione non e' mai verificata perche' l'espressione a sinistra vale false mentre quella a destra vale true per qualsiasi valore delle variabili a e b

Punteggio: 0

6. mancano dati per eseguire il calcolo

Punteggio: 0

Spiegazione:

Le espressioni a destra e sinistra sono coincidente: si applichi De Morgan.

Domanda n. 14

Argomento: Logica booleana

Peso: 1

Pubblica: N

Domanda: Relativamente all'espressione booleana seguente, dire quali delle seguenti

affermazioni sono vere:

$((!a \ \&\& \ !b) \ || \ (a \ \&\& \ b)) == ((!a \ || \ b) \ \&\& \ (a \ || \ !b))$

Risposte:

1. vale true qualsiasi siano i valori di a e b

Punteggio: 1

2. vale false qualsiasi siano i valori di a e b

Punteggio: 0

3. dipende da b: se b vale true, allora l'espressione vale false, altrimenti vale true

Punteggio: 0

4. dipende da b: se b vale true, allora l'espressione vale true, altrimenti vale false

Punteggio: 0

5. nessuna delle precedenti risposte e' corretta

Punteggio: 0

Spiegazione:

| a | b | !a&&!b | a&&b | (!a && !b) (a && b) |
|---|---|--------|------|------------------------|
| 0 | 0 | 1 | 0 | 1 |
| 0 | 1 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 |
| 1 | 1 | 0 | 1 | 1 |

| a | b | !a b | a !b | (!a b) && (a !b) |
|---|---|-------|-------|------------------------|
| 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 1 | 0 | 0 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 1 | 1 | 1 | 1 |

Domanda n. 15

Argomento: Iterazioni, array e stringhe

Peso: 1

Pubblica: N

Domanda: Quale frammento di codice e' equivalente al seguente ciclo while ?

```
int i = 0;  
while(i < k) i++;
```

Risposte:

1. int i = 0; do { i++; } while (i < k)

Punteggio: 0

2. int i; for(i = 0; i < k ; i++);

Punteggio: 1

3. int i = 0; while(i != k) i++;

Punteggio: 0

4. nessuna delle risposte precedenti e' corretta

Punteggio: 0

Spiegazione:

Si veda la definizione di ciclo for.

Domanda n. 16

Argomento: Iterazioni, array e stringhe

Peso: 1

Pubblica: N

Domanda: Quale frammento di codice e' equivalente al seguente ciclo for?

```
int i;  
for(i = 0; i < k; i++);
```

Risposte:

1. int i = 0; do { i++; } while (i < k);

Punteggio: 0

2. `int i = 0; if (i < k) do { i++; } while (i < k);`

Punteggio: 0

3. `int i = 0; while(i != k) i++;`

Punteggio: 0

4. `int i = 0; while(i < k) i++;`

Punteggio: 1

5. nessuna delle precedenti risposte e' corretta

Punteggio: 0

Spiegazione:

Si veda la definizione di ciclo for.

Domanda n. 17

Argomento: Iterazioni, array e stringhe

Peso: 1

Pubblica: N

Domanda: Quale frammento di codice e' equivalente al seguente ciclo do?

```
int i = 0;
int j = 0;
do {
    j = i++;
} while (i < n);
```

Risposte:

1. `int j = 0; for (int i = 0; i < n; i++) { j = i++; }`

Punteggio: 0

2. `int i = 0; int j = i++; while(i < n) { j = i++;}`

Punteggio: 1

3. `int i = 0; int j = 0; while(i < n) { j = i++; }`

Punteggio: 0

4. nessuna delle precedenti risposte e' corretta

Punteggio: 0

Spiegazione:

Si veda la definizione di ciclo do.

Domanda n. 18

Argomento: Variabili e tipi

Peso: 1

Pubblica: N

Domanda: Qual e' la sequenza corretta di istruzioni di assegnazione per scambiare i valori contenuti nelle due variabili m e n ?

Risposte:

1. `m = n; n = m;`

Punteggio: 0

2. `tmp = n; n = m; m = tmp;`

Punteggio: 1

3. `tmp = n; m = tmp; n = m;`

Punteggio: 0

4. `tmp = m; n = tmp; m=n;`

Punteggio: 0

5. nessuna delle precedenti risposte e' corretta

Punteggio: 0

Spiegazione:

Provare per credere!

Domanda n. 19

Argomento: Logica booleana

Peso: 1

Pubblica: N

Domanda: I due enunciati seguenti

```
int x; int y;  
...  
if (x / y > 0 && y != 0) ...  
if (y != 0 && x / y > 0) ...
```

Risposte:

1. sono equivalenti

Punteggio: 0

2. hanno un comportamento diverso quando y vale 0

Punteggio: 1

3. hanno un comportamento diverso quando x e' positivo

Punteggio: 0

4. Nessuna delle precedenti risposte e' corretta

Punteggio: 0

Spiegazione:

Il primo enunciato genera l'eccezione `ArithmeticException` quando y vale 0, il secondo non genera eccezioni quando $y = 0$ (cortocircuito!).

Domanda n. 20

Argomento: Iterazioni, array e stringhe

Peso: 1

Pubblica: N

Domanda: Il confronto lessicografico fra due stringhe richiede un numero di confronti tra singoli caratteri

Risposte:

1. almeno uguale alla lunghezza della stringa piu' lunga

Punteggio: 0

2. almeno uguale alla lunghezza della stringa piu' corta

Punteggio: 0

3. almeno uguale alla somma delle lunghezze delle due stringhe

Punteggio: 0

4. nessuna delle precedenti affermazioni e' corretta

Punteggio: 1

Spiegazione:

Il confronto lessicografico procede confrontando coppie di caratteri e termina quando i caratteri di una coppia sono diversi oppure quando sono stati esaminati tutti i caratteri di una delle due stringhe.

Ad esempio il confronto fra le stringhe "lupo" e "luna" richiede di confrontare tre caratteri: 'l'<->'l', 'u'<->'u', 'p'<->'n',

Il confronto fra le stringhe "lupo" e "lepre" richiede di confrontare 2 caratteri: 'l'<->'l', 'u'<->'e'.

Domanda n. 21

Argomento: Variabili e tipi

Peso: 1

Pubblica: N

Domanda: Qual e' l'effetto del seguente frammento di codice

```
String s = "21.0";  
if (s != null)  
{  
    double x = Double.parseDouble(s);  
    double y = x / 3.0;  
}
```

```
System.out.println("risultato: " + y);  
}
```

Risposte:

1. compila ed esegue correttamente, stampando la stringa "risultato: 7.0"

Punteggio: 1

2. errore in compilazione

Punteggio: 0

3. compila correttamente ma l'esecuzione provoca la generazione di un'eccezione

Punteggio: 0

4. nessuna delle precedenti

Punteggio: 0

Spiegazione:

Compila ed esegue senza generare eccezioni. Provare per credere!

Domanda n. 22

Argomento: Logica booleana

Peso: 1

Pubblica: N

Domanda: Quale dei seguenti enunciati esegue il confronto lessicografico fra due stringhe s1 e s2:

Risposte:

1. s1 == s2

Punteggio: 0

2. s1 = s2

Punteggio: 0

3. s1.equals(s2)

Punteggio: 0

4. `s1.compareTo(s2)`

Punteggio: 1

5. `s1.length() == s2.length()`

Punteggio: 0

6. nessuna delle precedenti risposte e' corretta

Punteggio: 0

Spiegazione:

L'ordine lessicografico viene stabilito dal metodo `compareTo()`

Domanda n. 23

Argomento: Iterazioni, array e stringhe

Peso: 1

Pubblica: N

Domanda: Il seguente frammento di codice in Java:

```
int n = 4;
int p = 1;
while (n > 0)
{
    p *= n;
}
System.out.println("risultato: " + p);
```

Risposte:

1. compila, ma in esecuzione ciclo all'infinito

Punteggio: 1

2. compila e in esecuzione invia a standard output il numero n^p , ovvero $4^1 = 4$

Punteggio: 0

3. compila e in esecuzione invia a standard output la somma dei numeri da 1 a 4

Punteggio: 0

4. non compila; il ciclo while deve essere sostituito con un ciclo for

Punteggio: 0

5. nessuna delle precedenti affermazioni e' corretta

Punteggio: 0

Spiegazione:

Cicla all'infinito perche' la variabile di ciclo n non viene mai decrementata.

Domanda n. 24

Argomento: Iterazioni, array e stringhe

Peso: 1

Pubblica: N

Domanda: Il seguente frammento di codice in Java:

```
int n = 4;
int p = 1;
while (n > 0)
{
    p *= n;
    n--;
}
System.out.println("risultato: " + p);
```

Risposte:

1. compila, ma in esecuzione cicla all'infinito

Punteggio: 0

2. compila e in esecuzione invia a standard output il prodotto dei numeri da 1 a 4

Punteggio: 1

3. compila e in esecuzione invia a standard output la somma dei numeri da 1 a 4

Punteggio: 0

4. non compila; il ciclo while deve essere sostituito con un ciclo for

Punteggio: 0

5. nessuna delle precedenti affermazioni e' corretta

Punteggio: 0

Spiegazione:

Calcola $4 \cdot 3 \cdot 2 \cdot 1$ ovvero $4!$.

Domanda n. 25

Argomento: Iterazioni, array e stringhe

Peso: 1

Pubblica: N

Domanda: Il seguente frammento di codice in Java:

```
int i = 0;
for(i = 0; i < k; i++)
    System.out.println("risultato: " + i);
```

Risposte:

1. compila, ma in esecuzione ciclo all'infinito

Punteggio: 0

2. compila e in esecuzione invia a standard output la somma dei numeri da 1 a k

Punteggio: 0

3. compila e in esecuzione invia a standard output il numero $k - 1$ se $k \geq 0$, il numero 0 se $k < 0$

Punteggio: 0

4. compila e in esecuzione invia a standard output il numero k se $k \geq 0$, il numero 0 se $k < 0$

Punteggio: 1

5. non compila; il ciclo for deve essere sostituito con un ciclo while

Punteggio: 0

6. nessuna delle precedenti affermazioni e' corretta

Punteggio: 0

Spiegazione:

e $k \leq 0$ il ciclo for non viene eseguito e il frammento stampa 0, se $k > 0$ i esce dal ciclo quando assume il valore k.