

1-) . while döngüsü, loop'u kontrol eden ve bir boole döndüren bir koşula sahiptir. Döngüye girmeden

önce kontrol edilir. Cevap : D

2-) traditional bir for döngüsü için, en iyi şekilde yukarı veya aşağı sayma değişkenine sahiptir.

Cevap : B

3-) Bir do-while döngüsü, döngünün yürütülmesinden sonra döngü durumunu kontrol eder. Bu

her zaman en az bir kez çalışmasını sağlar. Cevap : A

4-) For-each döngüsü her öğeden geçer ve bir değişken içerir. Cevap : C

5-) continue anahtar sözcüğü, döngü yinelenmesini hemen sonlandırmak ve yürütmeye devam etmek

için kullanılır. Cevap : B

6-) break anahtar kelimesi döngü yinelenmesini sonlandırmak, kalan tüm işlemleri atlamak için

kullanılır. Cevap : A

7-) . Traditional bir loop için parantez içinde 3 bölüm vardır. Cevap : B

8-) Geleneksel bir döngü ile dizinlerin kodda ziyaret edilme sırasını kontrol ederiz. Bu bir dizi içinde

artan veya azalan sırada döngü yapabileceğiniz anlamına gelir. Cevap : C

9-) Her biri için bir döngü ile döngü sırası sizin için belirlenir. Bir dizi ile bu, 0 indeksinden başlar.

Cevap : A

10-) Bir do-while döngüsü, döngünün sonunda bir boole döndüren bir koşula sahiptir. Cevap : A

11-) while döngüsü boole koşulu gerektirir. Bu nedenle, kod derlenmez. Cevap : B

12-) Kod derlendiğinde **cup,can**, sonucu çıkar.Cevap : B

14-) Kod derlendiğinde outputu **aa** olur. Cevap : A

15-) Kod sonsuz bir döngü üretir. Sürekli args çıktısı alınır. Cevap : D

16-) Kod derlendikten sonra output olarak 2 elde edilir. Cevap : B

17-) println() döngü içerisinde olmadığı için kod derlenemez. Cevap : C

18-) for döngüsünün içindeki üç parçanın da boş bırakılmasına izin verilir. Cevap : D

19-) Kod derlendiğinde **can, cup**, elde edilir. Cevap : A

- 21-) Tek bir ifade olduğu için parantez isteğe bağlıdır. Parantezlerin bir döngü gövdesini çevrelemesine izin verilir, bu nedenle kod derlenmez. Cevap : D
- 22-) B. for-each döngüsü sözdizimi olarak bir değişken ve iki nokta üst üste kullanır. Cevap : B
- 24-) Devam mektupları, ifadesi diğer iki ifadeyi yürütmeye devam eder. Bu nedenle, yalnızca ikinci ifade diyagramla doğru olarak eşleşir. Cevap : B
- 25-) while döngüsü, döngüye girmeden önce boole koşulunu kontrol eder. Bu kodda, koşul false olduğundan, döngü gövdesi hiçbir zaman çalıştırılmaz. Hiçbir çıktı üretilmez. Cevap : C
- 26-) Her biri için bir döngünün diziler ve ArrayList nesneleri ile kullanılmasına izin verilir. StringBuilder bu döngü için izin verilen bir tür değildir. Cevap : C
- 27-) Döngünün ilk yinelemesinde, if ifadesi şişirir- yürütür ve yazdırır. Ardından döngü durumu kontrol edilir. Değişken balloonInflated doğrudur, bu nedenle döngü koşulu yanlıştır ve döngü tamamlanır. Cevap : B
- 28-) İlk olarak döngü durumu kontrol edilir. Harfler 3 e eşit olmadığı için döngüye girilir. Döngü gövdesinde harfler ab olur ve harflerin uzunluğu 2 olur. Döngü tekrar kontrol edilir ve 2 döngü girilir ve abab ifadesi ile harflerin uzunluğu 4 olur. Sonra döngü tekrar kontrol edilir 4 3e eşit olmadığı için döngü gövdesine tekrar girilir. Bu bu şekilde sonsuza kadar devam eder. Cevap : D
- 29-) for döngüsünde initialization expression, boolean conditional, update statement . Cevap :B
- 30-) Kod derlendiğinde output 4 olur. Cevap : B
- 31-) i 10dan başlar. $10 > 3$ olur ve 3 çıkarır, i nin yeni değeri 7 olur. $7 > 3$, 7den 3 çıkarırız. $4 > 3$ olur ve i böylece 1 olur. k 1 yükseltilir. Döngü biter ve output 1 olur. Cevap : A
- 33-) Break ifadesi döngüde değil bu yüzden kod derlenemez. Cevap : C
- 34-) i ve j ifadeleri virgülle ayrılması gerekirken noktalı virgül ile ayrılmış. Kod hata verir. Cevap : C
- 36-) String tie ilk olarak null tanımlanmıştır. Daha sonra döngüye girdiğinde ise tie null ise “shoelace” olarak tanımlanır. Output da shoelace olur. Cevap : B
- 37-) Kod derlendiğinde output olarak x oluşur. Cevap : B

38-) Output olarak 4 oluşur. Cevap : C

39-) do-while döngüsü bir boolean koşulu gerektirir. StringBuilder bir boolean olmadığı için kod derlenemez. Cevap : C

40-) Derlendiğinde output 2 olarak görülür. Cevap : A

41-) Break t olursa her iki döngüden de çıkar. Cevap : C

42-) İki değişkeni başlatır ve her iki değişkeni de koşulda kullanır. iki dizinin boyutunu doğru kontrol ettiği için ilk iki öge kümesini yazdırır. Cevap : B

43-) Dış döngü iki kere yürütülür. İç döngü her biri için iki kere yürütüleceği için 4 kere yürütülmüş olur. Aşağıdaki şekilde outputu olur. Cevap : B

OCA OCA
OCA OCP
OCP OCA
OCP OCP

44-) Alpha ilk olarak çalışır ve java koşulunu kontrol eder. Beta döngünün yürütülmesini kontrol eder

ve beta yanlış döndürüldüğünden döngüye girilmez. Alpha, beta olur . Cevap : B

46-) Döngü koşulu açık olduğundan C 6 yinelemeden geçer. Ama diğer seçenekler 5 yinelemeden geçecektir. Cevap : C

47-) Döngü ilk kontrol edildiğinde tie nulldur. Ancak son durumda shoelace olmalı. Cevap : B

48-) for : while ifadesi yanlış olduğundan kod derlenemez. Cevap : C

49-) balloonInflated false olarak verilmiş ilk başta.Böylelikle if i.erisine girer ve true olur. Output inflate- olur. if artık tekrar çalışmaz. Cevap : D

50-) for döngüsünde, türün yalnızca bir kez belirtilmesine izin verilir. Bir virgül aynı ifadenin bir parçası olduğu için değişkenleri birden çok ayırır. Cevap : B

