

CORBEILLE GESTION DE LA MÉMOIRE

Tout afficher + Tout réduire -

SUJET

✓ Énoncé

EXERCICE : CORBEILLE GESTION DE LA MÉMOIRE

Question

1.1 - Comment est organisée la mémoire physique d'un ordinateur ?

Question

1.2 - Pouvez-vous citer les rôles de la MMU ?

Question

1.3 - Quels sont les mécanismes de gestion de la mémoire ?

Question

1.4 - A quoi sert le mécanisme de swap ?

Question

1.5 - A quoi servent les registres ? Et la mémoire cache ?

Question

1.6 - Quelles sont les deux structures de données utilisées par le framework .NET pour gérer et organiser la mémoire dans un programme ?

Question

1.7 - Pourquoi utiliser une pile pour la gestion de la mémoire et pas un tas pour tout ?

Question

1.8 - Avantages et inconvénients de 3 types de liaisons d'adresses ?

Question

1.9 - Pouvez-vous nommer des algorithmes d'allocation mémoire contiguë ? Donnez un exemple

Question

1.10 - Caractéristiques de la segmentation ?

Question

1.11 - Quel est l'inconvénient du mécanisme de segmentation "toute seule" ?

Question

1.12 - Que signifie l'expression "défaut de page" ?

Question

2.1 - Qu'appelle-t-on "exécution managée" ?

Question

2.2 - Décrivez le processus de gestion automatique de la mémoire

Question

2.3 - Décrivez le processus de libération de la mémoire

Question

2.4 - Qu'appelle-t-on les finaliseurs ?

Question

2.5 - Quel est l'intérêt de la méthode `GC.SuppressFinalize` ?

Question

2.6 - Que représente un délégué ?

Question

2.7 - Quelle est la différence entre un tableau de délégués et un délégué multicast ?

Question

2.8 - Complétez le code

```

1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 namespace @delegate
6 {
7     public delegate void _____(string musique);
8     public static class Dispatching
9     {
10         public static Frequence _____;
11         public static Frequence _1031;
12         private delegate _____ inK(Frequence f);
13         private static _____ inkD;
14         private static inK inkA;
15         private static inK inkMCA;
16         public static void abonnement(posteRadio poste, string station)
17         {
18             inkD = x => x -= _____;
19             inkA = x => _____ = new
20             Frequence(poste.ecouterMusique);
21             inkMCA = x => x += poste.ecouterMusique;
22             if (_____. _1066 != null)
23             {
24                 foreach (Frequence f in Dispatching._1066. _____
25                 ())
26                 {
27                     if (f== poste.ecouterMusique)
28                     {
29                         _____=inkD(_1066);
30                     }
31                 }
32             }
33             if (Dispatching._1031 != null)
34             {
35                 foreach (Frequence f in
36                 Dispatching._1031.GetInvocationList())
37                 {
38                     if (f == poste.ecouterMusique)
39                     {
40                         _____=inkD(_1031);

```

Question

2.9 - Complétez le code

```

1 using System;
2 using System.ComponentModel;
3 public class DisposeExample
4 {
5     public class MyResource: _____
6     {

```

```
7 private IntPtr handle;
8 private Component component = new Component() ;
9 private bool disposed = false;
10 public MyResource(IntPtr handle)
11 {
12     this.handle = handle;
13 }
14 public void Dispose()
15 {
16     Dispose(______);
17     GC. ____ (this);
18 }
19 protected ____ void Dispose(bool disposing)
20 {
21     if(!this. ____ )
22     {
23         if(____)
24         {
25             component. ____ ();
26         }
27         handle = IntPtr.Zero;
28         ____ = true;
29     }
30 }
31 [System.Runtime.InteropServices.DllImport("Kernel32")]
32 private extern static Boolean CloseHandle(IntPtr handle);
33
34 ~MyResource()
35 {
36     ____ (false);
37 }
38 }
39 public static void Main()
```