

# **NGÂN HÀNG 1000 CÂU HỎI TRẮC NGHIỆM NGÔN NGỮ LẬP TRÌNH C#**

**TRƯỜNG ĐẠI HỌC NAM CẦN THƠ  
GIẢNG VIÊN: Ths. Võ Văn Phúc**

**INTEGER DATA TYPES**

**1/Có bao nhiêu Byte được lưu trữ bởi Kiểu dữ liệu 'Long' trong C # .net?**

a) 8

b) 4

c) 2

d) 1

Đáp án: a

**2/Chọn tên ".NET class" từ kiểu dữ liệu "UInt" được tạo ra?**

a) System.Int16

b) System.UInt32

c) System.UInt64

d) System.UInt16

Đáp án: b

**3/Khai báo giá trị đúng cho các biến 'a' và 'b'?**

a) int a = 32, b = 40,6;

b) int a = 42; b = 40;

c) int a = 32; int b = 40;

d) int a = b = 42;

Đáp án: c

**4/Lỗi trong mã C # sau đây sẽ là gì?**

```
1.      Static Void Main ( Chuỗi [ ] args )
2.      {
3.      const int m = 100 ;
4.      int n = 10 ;
5.      const int k = n / 5 * 100 * n ;
6.      Bảng điều khiển . WriteLine ( m * k ) ;
7.      Bảng điều khiển . ReadLine ( ) ;
8.      }
```

a)'k' không được khai báo là hằng số.

b) Biểu thức gán cho 'k' phải là hằng số.

c) Biểu thức (m \* k) không hợp lệ.

d) 'm' được khai báo ở định dạng không hợp lệ.

Đáp án: b

**5/Sắp xếp kiểu dữ liệu sau theo thứ tự độ lớn tăng dần sbyte, short, long, int.**

a) dài <ngắn <int <sbyte

b) sbyte <ngắn <int <dài

c) ngắn <sbyte <int <dài

d) ngắn <int <sbyte <dài

Đáp án: b

**6/Kiểu dữ liệu nào nên được ưu tiên hơn để lưu trữ một số đơn giản như 35 để cải thiện tốc độ thực thi của chương trình?**

a) sbyte

b) ngắn

c) int

d) dài

Đáp án: a

**7/ Chức năng Chuyển đổi nào của 'Convert.ToInt32 ()' và 'Int32.Parse ()' là hiệu quả?**

i) Int32.Parse () chỉ được sử dụng cho chuỗi và ném ngoại lệ đối số cho chuỗi null.

ii) Convert.ToInt32 () được sử dụng cho các kiểu dữ liệu và trả về trực tiếp '0' cho chuỗi null.

a). ii

- b) Cả i, ii  
c) i  
d) Không có nội dung nào được đề cập

Đáp án: a

**8/ Cách đúng để gán giá trị cho biến 'c' khi int a = 12, float b = 3.5, int c.**

- a) `c = a + b;`  
b) `c = a + int (float (b));`  
c) `c = a + convert.ToInt32 (b);`  
d) `c = int (a + b);`

Đáp án: c

**9/ Bộ Mã C # đúng cho dữ liệu đã cho 'a' và 'b' để in đầu ra cho 'c' là 74?**

- a)  
`int a = 12 ;`  
`float b = 6 . 2f ;`  
`int c ;`  
`c = a / b + a * b ;`  
Bảng điều khiển . `WriteLine ( c ) ;`  
b)  
`int a = 12 ;`  
`float b = 6 . 2f ;`  
`int c ;`  
`c = a / chuyển đổi . ToInt32 ( b ) + a * b ;`  
Bảng điều khiển . `WriteLine ( c ) ;`

- c)  
`int a = 12 ;`  
`float b = 6 . 2f ;`  
`int c ;`  
`c = a / chuyển đổi . ToInt32 ( b ) + a * chuyển đổi . ToInt32 ( b ) ;`  
Bảng điều khiển . `WriteLine ( c ) ;`

- d)  
`int a = 12 ;`  
`float b = 6 . 2f ;`  
`int c ;`  
`c = chuyển đổi . ToInt32 ( a / b + a * b ) ;`  
Bảng điều khiển . `WriteLine ( c ) ;`

Đáp án: c

**10/ Sản lượng vẫn giống nhau hay khác nhau cho cả hai trường hợp?**

- i)  
`char l = 'k' ;`  
`float b = 19 . 0f ;`  
`int c ;`  
`c = ( l / chuyển đổi . ToInt32 ( b ) ) ;`  
Bảng điều khiển . `Writeline ( c ) ;`

- ii)  
`char l = 'k' ;`  
`float b = 19 . 0f ;`  
`int c ;`  
`c = Chuyển đổi . ToInt32 ( l / b ) ;`  
bàn điều khiển . `writeline ( c ) ;`

a) Có

b) Không

Đáp án: b

**11/ Loại số mặc định không có số thập phân là?**

- a) Int dài
- b) Int dài không dấu
- c) Int**
- d) Int không dấu

Đáp án: c

**12/ Đầu ra của đoạn mã C # sau đây sẽ là gì?**

- a) 23,453
- b) 22
- c) 23**
- d) 22,453

Đáp án: c

## FLOATING AND DECIMAL DATA TYPES

**13. Chọn một khai báo thuận tiện và khởi tạo một số dấu phẩy động:**

- a) float somevariable = 12.502D
- b) float somevariable = (Double) 12.502D
- c) float somevariable = (float) 12.502D**
- d) float somevariable = (Decimal)12.502D

Answer: c

Explanation: Chúng tôi không thể chuyển đổi ngầm một số "kép" trực tiếp sang bất kỳ kiểu dữ liệu nào khác. Ở đây, float của nó, chúng ta phải thêm kiểu dữ liệu bắt buộc vào số như:

float somevariable = (float)12.502D;

or

Double somevariable = (Double)12.502D;

**14. Số chữ số tối đa mà giá trị chính xác của kiểu dữ liệu float là hợp lệ?**

- a) Upto 6 digit
- b) Upto 8 digit
- c) Upto 9 digit
- d) Upto 7 digit**

Answer: d

Explanation: Theo định nghĩa.

**15. Kích thước hợp lệ của kiểu dữ liệu float là?**

- a) 10 Bytes
- b) 6 Bytes
- c) 4 Bytes**
- d) 8 Bytes

Answer: c

Explanation: Theo định nghĩa.

**16. Cách đúng để xác định giá trị 6.28 trong biến 'pi' trong đó giá trị không thể sửa đổi?**

- a) #define pi 6.28F
- b) pi = 6.28F
- c) const float pi = 6.28F**
- d) const float pi

pi = 6.28F

Answer: c

Explanation: Const là một từ khóa dự trữ bất cứ khi nào chúng được khai báo với bất kỳ biến nào. Giá trị được lưu trữ trong biến đó luôn luôn cố định. Bất kỳ sửa đổi nào được thực hiện để thay đổi giá trị của biến đó đều dẫn đến lỗi. Do đó, các tùy chọn #define pi 6.28F, pi = 6.28F và const float pi, pi = 6.28F bị từ chối vì giá trị không được khai báo là cố định. Bây giờ, đối với giá trị chỉ có tùy chọn 'const float pi = 6.28F' là đúng vì đối với giá trị trong khi khai báo một

biến hằng số, chúng ta cũng cần cung cấp một giá trị không đổi như được cung cấp trong const float pi = 6.28F. Vì vậy, tùy chọn const float pi = 6.28F là cách khai báo biến hằng số đúng.

**17. Tập hợp mã C # chính xác để hiển thị giá trị của biến 'c' đã cho là '25.302' sẽ là gì.**

a)

```
float a = (double) 12.502f;  
float b = 12.80f;  
float c;  
c = (float) a + b;  
Console.WriteLine(c);  
Console.ReadLine();
```

b)

```
float a = 12.502D;  
float b = 12.80f;  
float c;  
c = a + b;  
Console.WriteLine(c);  
Console.ReadLine();
```

**c)**

```
double a = 12.502;  
float b = 12.802f;  
float c;  
c = (float)a + b;  
Console.WriteLine(c);  
Console.ReadLine();
```

d)

```
double a = (float) 12.502f;  
float b = 12.80f;  
float c;  
c = a + b;  
Console.WriteLine(c);  
Console.ReadLine();
```

Answer: c

Explanation: Biểu hiện: c = (float) a + b.

**18. Phạm vi giá trị tối thiểu và tối đa được hỗ trợ bởi kiểu dữ liệu 'float' là?**

a)  $1.5 * 10^{-40} \rightarrow 3.4 * 10^{38}$

b)  $1.5 * 10^{-45} \rightarrow 3.4 * 10^{30}$

**c)  $1.5 * 10^{-45} \rightarrow 3.4 * 10^{38}$**

d)  $1.5 * 10^{-45} \rightarrow 3.4 * 10^{37}$

Answer: c

Explanation: Theo định nghĩa.

**19. Chọn sự khác biệt thích hợp giữa kiểu dữ liệu thập phân, float và double trong C#?**

i) Float and Double are floating binary point types while decimal is a floating decimal point type.

ii) Precision difference for float is '7' digit for double is '15' to '16' digit and for decimal is '28' to '29' digits.

iii) Một số giá trị không thể được đại diện chính xác do đó đối với các giá trị float và double là thích hợp hơn.

a) i

b) i, iii

**c) i, ii, iii**

d) ii, iii

Answer: c

Explanation: theo định nghĩa.

**20. Tại sao một biến float dừng tăng ở số '16777216' trong mã C # sau đây?**

```
float a = 0 ;  
while (true)  
{  
    a++;  
    if (a > 16777216)  
        break;  
}
```

a) Sign and Exponent for '16777217' is same as for '16777216'

**b) Mantissa is different for '16777216' and '16777217'**

c) Sign and Exponent for '16777217' is different from '16777216'

d) None of the mentioned

Answer: b

Explanation: 16777216 chính xác là 224 và sẽ được biểu diễn dưới dạng float 32-bit như sau:

dấu = 0 (số dương)

số mũ = 24 (được lưu trữ dưới dạng  $24 + 127 = 151 = 10010111$ )

phần định trị = 0

Dưới dạng biểu diễn dấu phẩy động 32 bit: 0 10010111 000000000000000000000000

Do đó: Giá trị =  $(+1) * 2^{24} * (1.0 + 0) = 2^{24} = 16777216$

Bây giờ, hãy xem số 16777217 hay chính xác là  $224 + 1$ :

dấu và số mũ giống nhau.

Mantissa phải chính xác là 2-24 để  $(+1) * 2^{24} * (1.0 + 2^{-24}) = 2^{24} + 1 = 16777217$

và đây là vấn đề thực tế. Phần định trị không thể có giá trị 2-24 vì nó chỉ có 23 bit, do đó số 16777217 không thể được biểu diễn với độ chính xác của các số dấu chấm động 32 bit.

**21. Đầu ra của mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
```

```
{  
    int x = 1;  
    float y = 2.4f;  
    short z = 1;  
    Console.WriteLine((float) x + y * z - (x + (short) y) );  
    Console.ReadLine();  
}
```

a) 0.4000004

b) 0.4000023

c) 0.0400021

**d) 0.4000001**

View Answer

Answer: d

Explanation: None.

Output : 0.4000001

**22. Một float chiếm 4 byte. Nếu giá trị tương đương thập lục phân của 4 byte này là A, B, C và D, thì khi float này được lưu trong bộ nhớ theo thứ tự nào sau đây, các byte này sẽ được lưu trữ?**

a) ABCD

b) DCBA

c) 0 \* ABCD

**d) Depends on big endian or little endian architecture**

Answer: d

Explanation: "Little Endian" có nghĩa là byte bậc thấp của số được lưu trong bộ nhớ ở địa chỉ thấp nhất và byte bậc cao ở địa chỉ cao nhất. Ví dụ: Số nguyên 4 byte

ABCD sẽ được sắp xếp trong bộ nhớ như sau:

Địa chỉ cơ sở + 0 Byte 0.

Địa chỉ cơ sở + 1 Byte 1.

Địa chỉ cơ sở + 2 Byte 2.

Địa chỉ cơ sở + 3 Byte 3.

Bộ xử lý Intel (những bộ xử lý được sử dụng trong PC) sử dụng thứ tự byte "Little Endian".

"Big Endian" có nghĩa là byte bậc cao của số được lưu trong bộ nhớ ở địa chỉ thấp nhất và byte bậc thấp ở địa chỉ cao nhất. Số nguyên 4 byte giống nhau sẽ được lưu trữ dưới dạng:

Địa chỉ cơ sở + 0 Byte 3.

Địa chỉ cơ sở + 1 Byte 2.

Địa chỉ cơ sở + 2 Byte 1.

Địa chỉ cơ sở + 3 Byte 0.

### 23. Giá trị mặc định của Kiểu dữ liệu Boolean là?

a) 0

b) True

**c) False**

d) 1

Answer: c

Explanation: theo định nghĩa.

### 24. Đầu ra của mã C # sau đây sẽ là gì?

```
public static void Main(string[] args)
{
    double ZERO = 0;
    Console.WriteLine("RESULT OF DIVISION BY ZERO IS :{0}", (0 / ZERO));
    Console.ReadLine();
}
```

a) 1

b) exception argument is thrown

**c) NaN**

d) 0

Answer: c

Explanation: None.

### 25. Công cụ chỉ định định dạng nào sau đây được sử dụng để in các giá trị thập lục phân và trả về giá trị đầu ra dưới dạng tương đương Hệ bát phân trong C #?

a) %hx for small case letters and %HX for capital letters

b) %x for small case letters and %X for capital letters

**c) No ease of doing it. C# don't provides specifier like %x or %O to be used with ReadLine() OR WriteLine(). We have to write our own function**

d) %Ox for small case letters and %OX for capital letters

Answer: c

Explanation: Không dễ dàng để làm điều đó. C # không cung cấp mã xác định như % x hoặc % O để được sử dụng với ReadLine () hoặc WriteLine (). Chúng ta phải viết hàm của riêng mình.

## CHAR TYPES AND STRING LITERALS

### 26. Kích thước của kiểu dữ liệu 'Char' là gì?

a) 8 bit

b) 12 bit

**c) 16 bit**

d) 20 bit

### 27. Đầu ra của đoạn mã C # sau đây sẽ là gì?

```
static void Main(string[] args)
{
    char c = 'g';
    string s = c.ToString();
    string s1 = "I am a human being" + c;
    Console.WriteLine(s1);
    Console.ReadLine();
}
```

- a) Tôi là một con người c
- b) Tôi là một con người**
- c) Tôi là một con người c
- d) Tôi là một con người

Giải thích: 'g' được lưu trữ trong biến ký tự 'c' mà sau này được chuyển đổi thành chuỗi bằng cách sử dụng phương thức Convert.ToString() và do đó được nối vào cuối chuỗi trong s1.

**28. Đã cho là mã ngày (ví dụ: "MTWTFSS") mà tôi cần tách và do đó tạo danh sách các ngày trong tuần theo chuỗi (ví dụ: "Thứ Hai", "Thứ Ba", "Thứ Tư", "Thứ Năm", "Thứ sáu Thứ Bảy Chủ nhật"). Một bộ mã được đưa ra cho mục đích này nhưng có lỗi xảy ra trong bộ mã đó liên quan đến việc chuyển đổi char thành chuỗi. Do đó, hãy chọn mã C# để giải quyết lỗi đã cho.**

```
static void Main(string[] args)
{
    var days = "MTWTFSS";
    var daysArray = days.ToCharArray().Cast<string>().ToArray();
    for (var i = 0; i < daysArray.Length; i++)
    {
        switch (daysArray[i])
        {
            case "M":
                daysArray[i] = "Monday";
                break;
            case "T":
                daysArray[i] = "Tuesday";
                break;
            case "W":
                daysArray[i] = "Wednesday";
                break;
            case "R":
                daysArray[i] = "Thursday";
                break;
            case "F":
                daysArray[i] = "Friday";
                break;
            case "S":
                daysArray[i] = "Saturday";
                break;
            case "U":
                daysArray[i] = "Sunday";
                break;
        }
    }
    daysArray[daysArray.Length - 1] = "and " + daysArray[daysArray.Length - 1];
    Console.WriteLine(string.Join(" ", daysArray));
}
```



```
}
```

- a) `var daysArray = new List<String>();`
- b) `var daysArray = days.Select(c => dayMapping[c]).ToArray();`
- c) `var daysArray = days.ToCharArray().Select(c => c.ToString()).ToArray();`
- d) `var daysArray = days.Select<String>();`

Bởi vì: Giải thích: Vấn đề nảy sinh do chuyển đổi kiểu từ "char" sang "string" vì một chuỗi không được kế thừa từ những người khác. Vì vậy, cách chuyển đổi nhanh chóng chỉ là sử dụng `Char.ToString ()`.

## 29. Đầu ra của đoạn mã C # sau đây sẽ là gì?

```
static void Main ( string [ ] args )
{
    {
        var dayCode = "MTWFS" ;
        var daysArray = new Danh sách < string > ( ) ;
        var list = new Dictionary < string , string >
        { { "M" , "Thứ Hai" } , { "T" , "Thứ Ba" } , { "W" , "Thứ Tư" } ,
          { "R" , "Thứ Năm" } , { "F" , "Thứ Sáu" } , { "S" , "Thứ Bảy" } ,
          { "U" , "Chủ nhật" }
        };
        for ( int i = 0 , max = dayCode . length ; i < max ; i ++ )
        {
            var tmp = dayCode [ i ] . ToString ( )
            if ( list . ContainsKey ( tmp ) )
            {
                ngàyArray . Thêm ( danh sách [ tmp ] ) ;
            }
        }
        Bảng điều khiển . WriteLine ( string . Join ( " \ n " , daysArray ) ) ;
    }
}
```

- a) Monday, Tuesday, Wednesday, Friday, Saturday, Sunday
- b)
  - Monday
  - Tuesday
  - Wednesday
  - Friday
  - Sunday

- c)
  - Monday
  - Tuesday
  - Wednesday
  - Friday
  - Saturday

- d) Monday, Tuesday, Wednesday, Friday, Saturday

Đáp án: c

Giải thích: Không có.

Đầu ra:

```
Thứ hai
Thứ ba
Thứ tư
```

Thứ sáu  
ngày thứ bảy

**30. Chọn sự khác biệt chính xác giữa kiểu dữ liệu char và varchar?**

Tôi. varchar không phải là unicode và char là kiểu dữ liệu ký tự unicode

ii. char là 'n' byte trong khi varchar là chiều dài thực tế tính bằng byte dữ liệu được nhập về kích thước lưu trữ

iii. varchar có độ dài thay đổi và char là chuỗi có độ dài cố định

iv. Đối với varchar, nếu một chuỗi nhỏ hơn độ dài tối đa thì nó được lưu trữ nguyên văn mà không có bất kỳ ký tự thừa nào trong khi đối với char nếu một chuỗi nhỏ hơn độ dài đã đặt, nó được đệm bằng các ký tự bổ sung để cân bằng độ dài của nó với độ dài đã cho

a) i, iii, iv

b) ii, iii, iv

c) i, ii, iv

d) iii, iv

**31. Phương thức String nào được sử dụng để so sánh hai chuỗi với nhau?**

a) Compare To()

b) Compare()

c) Copy()

d) ConCat()

b) Đáp án: b

Giải thích: So sánh () được sử dụng để so sánh hai chuỗi bằng cách lấy độ dài của chuỗi đang xét

**32. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    string s1 = "Delhi";
    string s2;
    s2 = s1.Insert(6, "Jaipur");
    Console.WriteLine(s2);
}
```

a) DelhJaipuri

b) Delhi Jaipur

c) Delhi

d) DelhiJaipur

Đáp án: d

Giải thích: Phương thức insert () của lớp string dùng để nối hai chuỗi s1 và s2.

Đầu ra:

DelhiJaipur

**32. Để hai chuỗi s1 và s2 bằng nhau, cách nào là đúng để tìm nội dung củ**

a. hai chuỗi bằng nhau?

b)

int c;

c = s1.CompareTo(s2);

c) if (s1 is s2)

d) if(strcmp(s1, s2))

Đáp án: b

Giải thích: Không có.

**33. Đầu ra của chuỗi C # sau đây sẽ là gì? (Nhập chuỗi: BOMBAY).**

```
static void Main(string[] args)
{
    string Str, Revstr = " ";
    int Length;
    Console.Write("Enter A String : ");
}
```

```

Str = Console.ReadLine();
Length = Str.Length - 1;
while (Length >= 0)
{
    Revstr = Revstr + Str[Length];
    Length --;
}
Console.WriteLine("Reverse String Is {0}", Revstr);
Console.ReadLine();

```

- a) BOMBA
- b) YABMOB**
- c) BOMAYB
- d) YABMO

Đáp án: b

Giải thích: Giải thích khái niệm đảo ngược của chuỗi mà không sử dụng bất kỳ phương thức sẵn có của chuỗi nào mà sử dụng điều kiện vòng lặp while.

Đầu ra: YABMOB

### 34. Chọn bộ mã C# thích hợp để chuyển đổi chuỗi sang dạng hexa.

```

static void Main(string[] args)
{
    string teststring = "MIKA@?&";
    string hex = ConvertstringToHex(teststring, system.Text.Encoding.Unicode);
    Console.WriteLine(hex);
}

```

a)

```

static string ConvertstringToHex(string input, system.Text.Encoding encoding)
{
    string Bytes = encoding.GetBytes(input);
    string Builder sbBytes = new Builder sbBytes();
    for each (byte 'b' in StringBytes) {
        sBytes.AppendFormat("{0:x2}", b);
    }
    return sbBytes.ToString();
}

```

b)

```

static string ConvertstringToHex(string input, system.Text.Encoding encoding)
{
    char[] Bytes = encoding.GetBytes(input);
    string Builder sbBytes = new Builder sbBytes(StringBytes.Length*2);
    for each (byte 'b' in StringBytes) {
        sBytes.AppendFormat("{0:X2}", b);
    }
    return sbBytes.ToString();
}

```

c)

```

public static string ConvertStringToHex(String input,
                                         System.Text.Encoding encoding)
{
    {
        Byte[] stringBytes = encoding.GetBytes(input);
        StringBuilder sbBytes = new StringBuilder(stringBytes.Length * 2);
        foreach (byte b in stringBytes)

```

```

    {
        sbBytes.AppendFormat("{0:X2}", b);
    }
    Console.WriteLine(sbBytes.ToString()); //sbBytes.ToString();
    return sbBytes.ToString();
}
}

```

d) None of the mentioned

Đáp án: c

Giải thích: Không có.

Đầu ra:

```

public static string ConvertStringToHex ( String input,
                                         System. Text . Encoding encoding ) {
    {
        Byte [ ] stringBytes = encoding . GetBytes ( đầu vào ) ;
        StringBuilder sbBytes = new StringBuilder ( stringBytes . Length * 2 ) ;
        foreach ( byte b trong stringBytes)
        {
            sbBytes . AppendFormat ( "{0: X2}" , b ) ;
        }
        Bảng điều khiển . WriteLine ( sbBytes . ToString ( ) ) ; //sbBytes.ToString (); trả về sbBytes .
        ToString ( ) ; }
    }
}

```

Bảng điều khiển . WriteLine ( sbBytes . ToString ( ) ) ; //sbBytes.ToString (); trả về sbBytes . ToString ( ) ; }

**35. Đoạn mã C # nào sau đây được sử dụng để chuyển đổi dạng hex sang dạng chuỗi?**

```

static void Main ( string [ ] args )
{
    string testString = " MIKA @ ? & ^" ;
    string normal = ConvertHexToString ( hex, System. Text . Encoding . Unicode ) ;
    Bảng điều khiển . WriteLine ( bình thường ) ;
    Bảng điều khiển . ReadLine ( ) ;
}

```

a)

```

chuỗi tĩnh công khai ConvertHexToString ( Chuỗi hexInput,
                                           Hệ thống. Văn bản . Mã hóa mã hóa )
{
    char [ ] numberChars = hexInput . Chiều dài ;
    byte [ ] byte = byte mới [ numberChars / 2 ] ;
    for ( int i = 0 ; i < numberChars ; i += 2 )
    {
        byte [ i / 2 ] = Chuyển đổi . ToByte ( hexInput . Chuỗi con ( i , 0 ) , 16 ) ;
    }
    trả về mã hóa . GetString ( byte ) ;
}

```

b)

```

chuỗi tĩnh công khai ConvertHexToString ( Chuỗi hexInput,
                                           Hệ thống. Văn bản . Mã hóa mã hóa )
{
    int numberChars = hexInput . Chiều dài ;
    byte [ ] byte = byte mới [ numberChars / 2 ] ;
    for ( int i = 0 ; i < numberChars ; i += 2 )
    {
        byte [ i / 2 ] = Chuyển đổi . ToByte ( hexInput . Chuỗi con ( i , 2 ) , 16 ) ;
    }
}

```

```

    }
    trả về mã hóa . GetString ( byte ) ;
}
c)
chuỗi tĩnh công khai ConvertHexToString ( Chuỗi hexInput,
    Hệ thống. Văn bản . Mã hóa mã hóa )

{

    string numberChars = hexInput . Chiều dài ;
    byte [ ] byte = byte mới [ numberChars ] ;
    for ( int i = 0 ; i < numberChars ; i += 2 )

    {
        byte [ i / 2 ] = Chuyển đổi . ToByte ( hexInput . Chuỗi con ( i , 2 ) , 16 ) ;
    }
    trả về mã hóa . GetString ( byte ) ;
}
d) Không có điều nào được đề cập

```

Đáp án: b

Giải thích: Không có.

Đầu ra:

```

public static string ConvertHexToString ( String hexInput,
    System. Text . Encoding encoding ) {
    int numberChars = hexInput . Chiều dài ;
    byte [ ] byte = byte mới [ numberChars / 2 ] ; for ( int i = 0 ; i < numberChars ; i
        += 2 )
    {
        byte [ i / 2 ] = Chuyển đổi . ToByte ( hexInput . Chuỗi con ( i , 2 ) , 16 ) ;
    }
    trả về mã hóa . GetString ( byte ) ; }

```

**36. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```

string s1 = "TÔI LÀ TỐT NHẤT" ;
chuỗi s2 ;
s2 = s1 .substring ( 5 , 4 ) ;
Console.WriteLine ( s2 ) ;

```

a) ĐÚNG NHẤT

b) TÔI LÀ TỐT NHẤT

c) TỐT NHẤT

d) TÔI LÀ

Đáp án: c

Giải thích: Chuỗi con () của lớp chuỗi dùng để trích xuất chuỗi con từ chuỗi đã cho. Trong điều kiện chuỗi con đã cho, nó trích xuất một chuỗi con bắt đầu ở vị trí thứ 5 và kết thúc ở vị trí thứ 4.

Đầu ra: TỐT

**37. Phát biểu đúng về chuỗi là?**

a) một chuỗi được tạo trên ngăn xếp

b) một chuỗi có bản chất nguyên thủy

c) một chuỗi được tạo trên heap

d) được tạo ra từ chuỗi trên một ngăn xếp hoặc trên một đồng phụ thuộc vào độ dài của chuỗi

Đáp án: c

Giải thích: Không có.

**38. Nguyên văn chuỗi chữ được sử dụng tốt hơn cho?**

- a) Sự thuận tiện và khả năng đọc chuỗi tốt hơn khi văn bản chuỗi bao gồm các ký tự dấu gạch chéo ngược
- b) Được sử dụng để khởi tạo các chuỗi nhiều dòng
- c) Để nhúng dấu ngoặc kép bằng cách sử dụng dấu ngoặc kép bên trong một chuỗi nguyên văn
- d) Tất cả các điều được đề cập

Đáp án: d

Giải thích: Theo định nghĩa.

**39. Tại sao các chuỗi có kiểu tham chiếu trong C # .NET?**

- a) Để tạo chuỗi trên ngăn xếp
- b) Để giảm kích thước của chuỗi
- c) Để khắc phục sự cố của stackoverflow
- d) Không có cách nào được đề cập

Đáp án: b

Giải thích: Vấn đề tràn ngăn xếp rất dễ xảy ra vì giao thức truyền tải được sử dụng trên web ngày nay là 'HTTP' và tiêu chuẩn dữ liệu là 'XML'. Do đó, cả hai đều sử dụng rộng rãi các chuỗi sẽ dẫn đến vấn đề tràn ngăn xếp. Vì vậy, để tránh trường hợp này, tốt hơn hết bạn nên đặt các chuỗi thành một kiểu tham chiếu và do đó tạo nó trên heap.

**INITIALIZATION OF VARIABLES****40. Đầu ra của mã C # sau đây sẽ là gì?**

```
1. static void Main(string[] args)
2. {
3.     int a = 5;
4.     int b = 10;
5.     int c;
6.     Console.WriteLine(c = ++ a + b ++);
7.     Console.WriteLine(b);
8.     Console.ReadLine();
9. }
```

- a) 11, 10
- b) 16, 10
- c) 16, 11
- d) 15, 11

[View Answer](#)

**41. Vị trí lưu trữ được sử dụng bởi bộ nhớ máy tính để lưu trữ dữ liệu cho một ứng dụng sử dụng là?**

- a) Con trỏ
- b) Hằng số
- c) Biến
- d) Không có điều nào được đề cập

[View Answer](#)

**42. Sự khác biệt giữa từ khóa 'var' và 'dynamic'?**

- a) 'Var' được giới thiệu trong C # (3.0) và 'Dynamic' được giới thiệu trong C # (4.0)
- b) 'Var' là một loại biến nơi khai báo được thực hiện tại thời điểm biên dịch bằng trình biên dịch trong khi khai báo 'Động' được thực hiện tại thời gian chạy bằng trình biên dịch
- c) Đối với Lỗi 'Var' bị bắt tại thời gian biên dịch và đối với Lỗi 'Động' bị bắt trong thời gian chạy
- d) Tất cả những điều đã đề cập

[View Answer](#)

**43. Các mã C # sau đây là?**

```
1. Myclass class;  
   Myclass class2 = null;  
2. int i;  
   int j = 0;
```

- a) Đúng cho (1); Sai cho (2)
- b) Đúng cho (2); Sai cho (1)
- c) Cả (1) và (2) là tương đương
- d) Cả (1) và (2) không phải là tương đương

[View Answer](#)

**44. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
int a,b;  
a = (b = 10) + 5;
```

- a) b = 10, a = 5
- b) b = 15, a = 5
- c) a = 15, b = 10
- d) a = 10, b = 10

[View Answer](#)

**45. Đầu ra của chuyển đổi mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)  
{  
    char a = 'A';  
    string b = "a";  
    Console.WriteLine(Convert.ToInt32(a));  
    Console.WriteLine(Convert.ToInt32(Convert.ToChar(b)));  
    Console.ReadLine();  
}
```

- a) 1, 97
- b) 65, 97
- c) 65, 97
- d) 97, 1

[View Answer](#)

**46. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)  
{  
    String name = "Dr.Gupta";  
    Console.WriteLine("Good Morning" + name);  
}
```

- a) Dr.Gupta
- b) Chào buổi sáng
- c) Chào buổi sáng Dr.Gupta
- d) Tên Chào buổi sáng

[View Answer](#)

**47. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)  
{  
    int a = 5;  
    int b = 10;  
    int c;  
    Console.WriteLine(c = a-- - ++b);  
    Console.WriteLine(b);  
    Console.ReadLine();  
}
```

- ```
}
```
- a) -7, 10  
b) -5, 11  
c) -6, 11  
d) 15, 11

[View Answer](#)

**48. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    const int a = 5;
    const int b = 6;
    for (int i = 1; i <= 5; i++)
    {
        a = a * i;
        b = b * i;
    }
    Console.WriteLine(a);
    Console.WriteLine(b);
    Console.ReadLine();
}
```

- a) 600, 720  
b) Lỗi thời gian biên dịch  
c) 25, 30  
d) 5, 6

[View Answer](#)

**49. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    string Name = "He is playing in a ground.";
    char[] characters = Name.ToCharArray();
    StringBuilder sb = new StringBuilder();
    for (int i = Name.Length - 1; i >= 0; --i)
    {
        sb.Append(characters[i]);
    }
    Console.Write(sb.ToString());
    Console.ReadLine();
}
```

- a) Anh ấy đang chơi trong một con cá mú  
b). nền a đang chơi là He  
c) .dnurag a ni gniyalp si eH  
d) Anh ấy chơi mộ

[View Answer](#)

## SCOPE AND LIFETIME OF VARIABLES

**50. Chọn đúng loại phạm vi biến cho các biến được định nghĩa trong C # sau đây.**

```
class ABC
{
    static int m;
    int n;
```



```
void fun (int x , ref int y, out int z, int[] a)
{
    int j = 10;
}
```

a) m = static variable, n = local variable, x = output parameter, y = reference parameter, j = instance variable, z = output parameter, a[0] = array element

b) m = static variable, n = instance variable, x = value parameter, y = reference parameter, j = local variable, z = output parameter, a[0] = array element

c) m = static variable, n = instance variable, x = reference parameter, y = value parameter, j = local variable, z = output parameter, a[0] = array element

d) m = local variable, n = instance variable, x = reference parameter, y = value parameter, j = static variable, z = output parameter, a[0] = array element

**51. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
class Program
{
    static void Main(string[] args)
    {
        int i;
        for (i = 0; i < 5; i++)
        {
            Console.WriteLine(i);
        }
        Console.ReadLine();
    }
}
```

a) 0, 1, 2, 3, 4, 5

b) 0, 1, 2, 3

c) 0, 1, 2, 3, 4

d) 0, 0, 0, 0, 0

**52. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
class Program
{
    static void Main(string[] args)
    {
        int i;
        for ( i = 0; i < 5; i++)
        {
        }
        Console.WriteLine(i);
        Console.ReadLine();
    }
}
```

a) 0, 1, 2, 3, 4, 5

b) 0, 1, 2, 3, 4

c) 5

d) 4

**53. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
class Program
{
    static void Main(string[] args)
    {
        int i;
```

```

    for ( i = 0; i < 5; i++)
    {
        int j = 0;
        j += i;
        Console.WriteLine(j);
    }
    Console.WriteLine(i);
    Console.ReadLine();
}
}

```

a) 0, 1, 2, 3, 4, 5, 6

b) 0, 1, 2, 3, 4, 5

c) 0, 1, 2, 3, 4

d) 0, 1, 2, 3

**54. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```

static void Main(string[] args)
{
    int i ;
    for (i = 0; i < 5; i++)
    {
        int j = 0;
        j += i;
        Console.WriteLine(j);
    }
    Console.WriteLine( i * j);
    Console.ReadLine();
}

```

a) 0, 1, 6, 18, 40

b) 0, 1, 5, 20, 30

c) Compile time error

d) 0, 1, 2, 3, 4, 5

**55. Phạm vi của biến có liên quan đến định nghĩa của biến như:**

i. Region of code within which variable value is valid and hence can be accessed.

ii. No, relation with region where variable is declared its value is valid in entire scope.

a) i

b) ii

c) i, ii

d) None of the mentioned

**56. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```

class Program
{
    public static void Main(string[] args)
    {
        int i = 100;
        for (a = 0; a < 5; a++)
        {
            int i = 200;
            Console.WriteLine(a * i);
        }
        Console.ReadLine();
    }
}

```

a) 5, 10, 15, 20

b) 0, 5, 10, 20

c) Compile time error

d) 0, 1, 2, 3, 4

**57. Cú pháp khai báo và khởi tạo biến dữ liệu là?**

a) <data type><var\_name> = <Value>;

b) <data type><var\_name>;

c) <var\_name><data type>;

d) <var\_name> = <value>;

**58. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

`class Program`

```
{  
    public static void Main(string[] args)  
    {  
        int i, j;  
        i = (j = 5) + 10;  
        Console.WriteLine(i);  
        Console.WriteLine(j);  
        Console.ReadLine();  
    }  
}
```

a) 15, 15

b) 10, 5

c) 15, 5

d) 10, 15

**59. Chọn sự khác biệt hiệu quả giữa 'Boxing' và 'Unboxing'.**

a) 'Boxing' is the process of converting a value type to the reference type and 'Unboxing' is the process of converting reference to value type

b) 'Boxing' is the process of converting a reference type to value type and 'Unboxing' is the process of converting value type to reference type

c) In 'Boxing' we need explicit conversion and in 'Unboxing' we need implicit conversion

d) Both 'Boxing' and 'Unboxing' we need implicit conversion

**60. Chọn sự khác biệt giữa loại tham chiếu và loại giá trị:**

i. Memory allocated to 'Value type' is from heap and reference type is from 'System. ValueType'

ii. Memory allocated to 'Value type' is from 'System. ValueType' and reference type is from 'Heap'

iii. Structures, enumerated types derived from 'System. ValueType' are created on stack, hence known as Value type and all 'classes' are reference type because values are stored on heap

a) i, iii

b) ii, iii

c) i, ii, iii

d) i

**61. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
public static void Main(string[] args)  
{  
    int i = 123;  
    object o = i;  
    i = 456;  
    System.Console.WriteLine("The value-type value = {0}", i);  
    System.Console.WriteLine("The object-type value = {0}", o);  
    Console.ReadLine();  
}
```

a) 123, 123

b) 456, 123

c) 456, 456

d) 123, 456

**62. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
public static void Main(string[] args)
{
    int i = 546;
    object o = i;
    int n = (int) o;
    o = 70;
    System.Console.WriteLine("The value-type value = {0}", n);
    System.Console.WriteLine("The object-type value = {0}", o);
    Console.ReadLine();
}
```

a) 546, 0

b) 546, 546

c) 546, 70

d) 70, 546

### TYPE CONVERSION IN EXPRESSIONS

**63. Sự cần thiết đối với 'Chuyển đổi kiểu dữ liệu' trong C # là gì?**

a) To store a value of one data type into a variable of another data type

b) To get desired data

c) To prevent situations of runtime error during change or conversion of data type

d) None of the mentioned

[View Answer](#)

**Answer: c**

Explanation: Theo định nghĩa.

**64. Các loại 'Chuyển đổi dữ liệu' trong C #?**

a) Implicit Conversion

b) Explicit Conversion

c) Implicit Conversion and Explicit Conversion

d) None of the mentioned

[View Answer](#)

**Answer: b**

Explanation: Theo định nghĩa

**65. 'Chuyển đổi ngầm định' tuân theo thứ tự chuyển đổi theo khả năng tương thích của loại dữ liệu như:**

a) float < char < int

b) char < int < float

c) int < char < float

d) float < int < char

[View Answer](#)

**Answer: b**

Explanation: Không ai.

**66. Đối với mã C # sau, hãy chọn giải pháp thích hợp để chuyển đổi kiểu dữ liệu.**

```
static void Main(string[] args)
{
    int num1 = 20000;
    int num2 = 50000;
    long total;
```

```
total = num1 + num2;  
Console.WriteLine("Total is : " +total);  
Console.ReadLine();  
}
```

- a) Compiler will generate runtime error
- b) Conversion is implicit type, no error generation**
- c) Specifying data type for conversion externally will solve the problem
- d) None of the mentioned

[View Answer](#)**Answer: b**

Explanation: Vì, chuyển đổi kiểu dữ liệu là kiểu ngầm định vì 'int' là một tập con của 'longtype' do đó không cần phải chuyển đổi dữ liệu một cách rõ ràng từ kiểu này sang kiểu khác. Trình biên dịch sẽ tự động thực hiện chuyển đổi.

Output: Total is: **70000**.

**67. Tập hợp con của kiểu dữ liệu 'int' là \_\_\_\_\_**

- a) long, ulong, ushort
- b) long, ulong, uint
- c) long, float, double**
- d) long, float, ushort

[View Answer](#)**Answer: c**

Explanation: Theo định nghĩa.

**68. Kiểu chuyển đổi mà trình biên dịch không thể chuyển đổi kiểu dữ liệu một cách ngầm định là?**

- a) ushort to long
- b) int to uint**
- c) ushort to long
- d) byte to decimal

[View Answer](#)**Answer: b**

Explanation: 'Int' là số nguyên 32 bit có dấu trong khi 'uint' là số nguyên không dấu 32 bit.

Phạm vi của int lớn hơn uint. Vì vậy, trình biên dịch không thể chuyển đổi hoàn toàn từ kiểu dữ liệu lớn hơn sang kiểu dữ liệu nhỏ hơn.

**69. Nhược điểm của Chuyển đổi rõ ràng là?**

- a) Makes program memory heavier
- b) Results in loss of data**
- c) Potentially Unsafe
- d) None of the mentioned

[View Answer](#)**Answer: b**

Explanation: Theo định nghĩa.

**70. Đối với bộ mã C # đã cho, có thể chuyển đổi không?**

```
static void Main(string[] args)  
{  
    int a = 76;  
    char b;  
    b = (char)a;  
    Console.WriteLine(b);  
    Console.ReadLine();  
}
```

- a) Compiler will generate runtime error
- b) Conversion is explicit type**

- c) Compiler will urge for conversion from 'integer' to 'character' data type  
d) None of the mentioned

[View Answer](#)

**Answer: b**

Explanation: Vì, chuyển đổi nhất định thuộc kiểu rõ ràng vì một kiểu dữ liệu ở dạng số nguyên và kiểu dữ liệu khác ở dạng 'char'. Trình biên dịch là cần thiết để phân biệt rõ ràng giữa cả hai loại kiểu dữ liệu và do đó, người ta cần chỉ định rõ ràng kiểu dữ liệu vì trình biên dịch không thể thực hiện chuyển đổi tự động.

Output : L.

### 71. Đầu ra của mã C # sau đây sẽ là gì?

```
1. static void Main(string[] args)
2. {
3.     float sum;
4.     int i;
5.     sum = 0.0F;
6.     for (i = 1; i <= 10; i++)
7.     {
8.         sum = sum + 1 / (float)i;
9.     }
10.    Console.WriteLine("sum =" + sum);
11.    Console.ReadLine();
12. }
```

a) 2.000

b) 2.910

**c) 2.928**

d) 3.000

[View Answer](#)

**Answer: c**

Explanation: Không ai.

Output :

sum = 2.928698.

### 72. Chuyển đổi nào hợp lệ cho mã C # sau đây?

```
static void Main(string[] args)
{
    int a = 22;
    long b = 44;
    double c = 1.406;
    b = a;
    c = a;
    a = b;
    b = c;
}
```

**a) c = a, b = c**

b) a = c, b = a

c) b = a, c = a

d) All of the mentioned

[View Answer000](#)

**Answer: a**

Explanation: Việc chuyển đổi kiểu dữ liệu từ 'int' thành 'double' có tính chất ngầm định đối với 'c = a' vì int là tập con của double nhưng không áp dụng tương tự cho 'b = c' vì khi đó 'c' có phạm vi dài dữ liệu rộng hơn 'b' để chuyển đổi rõ ràng là cần thiết.

Output :

Error 1: Can not implicitly convert type 'long' to 'int'. An explicit conversion exists (are you missing a cast?).

Error 2: Cannot implicitly convert type 'double' to 'long'. An explicit conversion exists (are you missing a cast?).

Correct solution :

```
static void Main(string[] args)
{
    int a = 22;
    long b = 44;
    double c = 1.406;
    b = a;
    c = a;
    a = (int)b;
    b = (long)c;
}
```

### ARITHMETIC OPERATORS

73. Đầu ra của đoạn mã C # sau đây sẽ là gì?

```
static void Main(string[] args)
{
    float a = 16.4f;
    int b = 12;
    float c;
    c = a * ( b + a ) / ( a - b );
    Console.WriteLine("result is : " + c);
    Console.ReadLine();
}
```

- a) 106
- b) 104.789
- c) 105.8546**
- d) 103.45

[View Answer](#)

Trả lời: c

Giải thích: Biểu thức đầu tiên được đánh giá là 'b + a' vì cả hai được kết hợp. Tiếp theo, biểu thức được nhân với toán hạng 'a' tức là a (b + a) toàn bộ kết quả của tử số được cộng lại và chia cho biểu thức mẫu số (a - b).

Đầu ra:

result is : 105.8546.

74. Đầu ra của mã C # sau đây sẽ là gì?

```
static void Main(string[] args)
1. {
2.     int a, b, c, x;
3.     a = 90;
4.     b = 15;
5.     c = 3;
6.     x = a - b / 3 + c * 2 - 1;
7.     Console.WriteLine(x);
8.     Console.ReadLine();
9. }
```

- a) 92
- b) 89
- c) 90**

d) 88

[View Answer](#)

Trả lời: c

Giải thích: Quá trình đánh giá cơ bản bao gồm hai lần chuyển từ trái sang phải thông qua biểu thức. Trong lần vượt qua đầu tiên, các toán tử ưu tiên cao được áp dụng và trong lần vượt qua thứ hai, các toán tử ưu tiên thấp được áp dụng khi chúng gặp phải.

First pass :

step 1 :  $x = 90 - 15 / 3 + 3 * 2 - 1$  (15 / 3 evaluated)step 2 :  $x = 90 - 5 + 3 * 2 - 1$ step 3 :  $x = 90 - 5 + 3 * 2 - 1$  (3 \* 2 is evaluated)step 4 :  $x = 90 - 5 + 6 - 1$ 

Second pass :

step 5 :  $x = 85 + 6 - 1$  (90 - 5 is evaluated)step 6 :  $x = 91 - 1$  (85 + 6 is evaluated)step 7 :  $x = 90$  (91 - 1 is evaluated)

Output : 90.

**75. Đầu ra của mã C # sau đây sẽ là gì?** to Install Unity on Ubuntu 18.04 [Complete[Procedure\]](#)

```
static void Main(string[] args)
{
    int a, b, c, x;
    a = 80;
    b = 15;
    c = 2;
    x = a - b / (3 * c) * (a + c);
    Console.WriteLine(x);
    Console.ReadLine();
}
```

a) 78

**b) -84**

c) 80

d) 98

[View Answer](#)

Đáp án: b

Giải thích: Bất cứ khi nào dấu ngoặc đơn được sử dụng, các biểu thức trong dấu ngoặc đơn sẽ có mức độ ưu tiên cao hơn. Nếu Hai hoặc nhiều tập hợp các dấu ngoặc đơn xuất hiện lần lượt như được hiển thị ở trên, biểu thức chứa ở bên trái được đánh giá trước và bên phải sau cùng.

First pass:

Step 1:  $80 - 15 / (3 * 2) * (80 + 2)$ Step 2:  $80 - 15 / 6 * 82$  ((3 \* 2) evaluated first and (80 + 2) evaluated later)

Second pass:

Step 3:  $80 - 2 * 82$ Step 4:  $80 - 164$ .

Third pass:

Step 5 : -84. (80 - 164 is evaluated)

Output : -84 .

**76.Thứ tự ưu tiên chính xác là:**

a) '/'&gt; '%'&gt; '\*'&gt; '+'

b) '/'&gt; '\*'&gt; '%'&gt; '+'

**c) '\*'> '/'> '%'> '+'**

d) '%'&gt; '\*'&gt; '/'&gt; '+'



[View Answer](#)

Trả lời: c

Giải thích: Theo định nghĩa.

**77. Đầu ra của mã C # sau đây sẽ là gì?**

```
int i, j = 1, k;  
1. for (i = 0; i < 3; i++)  
2. {  
3.     k = j++ - ++j;  
4.     Console.Write(k + " ");  
5. }
```

a) -4 -3 -2

b) -6 -4 -1

**c) -2 -2 -2**

d) -4 -4 -4

[View Answer](#)

Trả lời: c

Giải trình:

Here i = 0, j = 1.

k = 1 - 3 ( j++ = 2 and ++j = 3 )  
k = -2.  
i = 1, j = 3.  
k = 3 - 5 ( j++ = 4 and ++j = 5 )  
k = -2.  
i = 2, j = 5.  
k = 5 - 7 ( j++ = 6 and ++j = 7 )  
k = -2.

Output : -2 , -2 , -2.

**78. Đầu ra của mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
```

```
{  
    int b= 11;  
    int c = 7;  
    int r = 5;  
    int e = 2;  
    int l;  
    int v = 109;  
    int k;  
    int z,t,p;  
    z = b * c;  
    t = b * b;  
    p = b * r * 2;  
    l = (b * c) + (r * e) + 10;  
    k = v - 8;  
    Console.WriteLine(Convert.ToString(Convert.ToChar(z)) + " " +  
        Convert.ToString(Convert.ToChar(t)) + Convert.ToString(Convert.ToChar(p)) +  
        Convert.ToString(Convert.ToChar(l)) + Convert.ToString(Convert.ToChar(v)) +  
        Convert.ToString(Convert.ToChar(k)));  
    Console.ReadLine();  
}
```

a) My Name

b) My nAme

**c) My name**

d) Myname

[View Answer](#)

Trả lời: c

Giải thích: Giải biểu thức  $l = (b * c) + (r * e) + 10$ . Trong khi từ trái sang phải dấu ngoặc được ưu tiên trước.

Step 1 :  $b * c$  is evaluated first inside first parentheses.

Step 2 :  $r * e$  is evaluated second on right side of first addition symbol.

Step 3 : After evaluating both parentheses 10 is added to value of both.

Output : My name.

**79. Đầu ra của mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    int n = 5;
    int x = 4;
    int z, c, k;
    for (c = 1; c <= n; c++)
    {
        for (k = 1; k <= c; k++)
        {
            z = 3 * x * x + 2 * x + 4 / x + 8;
            Console.Write(Convert.ToString(Convert.ToChar(z)));
        }
        Console.WriteLine("\n");
    }
    Console.ReadLine();
}
```

|                              |                              |                                              |                              |
|------------------------------|------------------------------|----------------------------------------------|------------------------------|
| a)<br>A<br>AA<br>AAA<br>AAAA | b)<br>A<br>AB<br>ABC<br>ABCD | <b>c)</b><br>A<br>AA<br>AAA<br>AAAA<br>AAAAA | d)<br>A<br>BC<br>DEF<br>DEFG |
|------------------------------|------------------------------|----------------------------------------------|------------------------------|

[View Answer](#)

Trả lời: c

Giải thích: Giải biểu thức cho giá trị của 'z' as 65. Với, mỗi đoạn giá trị vòng lặp, số giá trị của 'z' tăng lên cho mỗi hàng là

Row 1: A

Row 2: AA

-

-

Row 5: AAAAA

Output : A

AA

AAA

AAAA

AAAAA

**80. Đầu ra của mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
```

```

{
    int n = 5;
    int x = 4;
    int z, c, k;
    z = 3 * x * x + 2 * x + 4 / x + 8;
    for (c = 1; c <= n; c++)
    {
        for (k = 1; k <= c; k++)
        {
            Console.Write(Convert.ToString(Convert.ToChar(z)));
            z++;
        }
        Console.WriteLine("\n");
    }
    Console.ReadLine();
}

```

|                                       |                                       |                                              |                                    |
|---------------------------------------|---------------------------------------|----------------------------------------------|------------------------------------|
| a)<br>A<br>AA<br>AAA<br>AAAA<br>AAAAA | b)<br>A<br>AB<br>ABC<br>ABCD<br>ABCDE | <b>c)</b><br>A<br>BC<br>DEF<br>GHIJ<br>KLMNO | d)<br>A<br>AB<br>BC<br>BCD<br>BCDE |
|---------------------------------------|---------------------------------------|----------------------------------------------|------------------------------------|

[View Answer](#)

Trả lời: c

Giải thích: Giải giá trị của biểu thức 'z' là 65. Trước khi vào bên trong vòng lặp đầu tiên

Bước 1: c = 1, n = 5

k = 1, k <= 1. (như c = 1)

z = 65 được chuyển đổi thành 'A' vì giá trị ascii của 'A' là 65.

z ++ tăng cho điều kiện lặp tiếp theo bằng '1' as 66.

Hàng 1: A

Bước 2: c = 2, n = 5

k = 2, k <= 2. (như c = 2)

z = 66 từ bước 1 giá trị được chuyển đổi của 66 là 'B'. Vì, k <= 2

vòng lặp sẽ lặp lại đến giá trị thứ hai sau 66 là 67 như z

tăng từ 66 lên +1 là '67'. Vì vậy, chuyển đổi giá trị ascii của 67 thành ký tự là 'C'.

Hàng 2: B C

Tương tự,

Bước 3:

Hàng 3: D E F

Bước 4:

Hàng 4: G H I J

Bước 5:

Hàng 5: K L M N O.

Output: A

BC

DEF

GHIJ

KLMNO

**81. Đầu ra của mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
```

```
{
```

```
    int a, b;
```

```
    int c = 10;
```

```

int d = 12;
int e = 5;
int f = 6;
a = c * (d + e) / f + d;
Console.WriteLine(a);
b = c * (d + e / f + d);
Console.WriteLine(b);
if (a < b)
{
    Console.WriteLine(" parentheses changes values");
}
else if (a > b)
{
    Counterintelligence("they have same value");
}
Console.ReadLine();
}

```

- a) Chúng có cùng giá trị  
**b) Dấu ngoặc đơn thay đổi giá trị**  
c) Vì cả hai đều có giá trị bằng nhau nên không kết luận  
d) Không có điều nào được đề cập

[View Answer](#)

Đáp án: b

Giải thích: Việc giải cho biểu thức 'a' bên trong dấu ngoặc đơn được ưu tiên đánh giá (d + e) là 17.

a = 10 \* 17/6 + 12.

a = 40.

Solving for expression 'b' expression inside parentheses (d + e /f + d) are evaluated as (12 + 5/6 + 12)

b = 10 \*(12 + 5/6 + 12).

b = 240.

Output : 40

240 parentheses changes values.

## 82. Cách đúng để tăng các toán tử là:

- a) ++ a ++  
b) b ++ 1  
c) c + = 1  
d) d = + 1

Xem câu trả lời

Trả lời: c

Giải thích: Dấu += này được gọi là toán tử bàn tay ngắn, giống như biến = biến +1. Tương tự, a- = 1 là a = a-1, a \* = 1 là a = a \* 1. Chúng được sử dụng để làm cho mã ngắn và hiệu quả.

## BITWISE AND CONDITIONAL OPERATORS

### Toán tử Bitwise và Điều kiện

## 83. Đầu ra của đoạn mã C # sau đây sẽ là gì?

```
static void Main(string[] args)
```

```
{
```

```
    byte varA = 10;
```

```
    byte varB = 20;
```

```
    long result = varA & varB;
```

```
    Console.WriteLine("{0} AND {1} Result :{2}", varA, varB, result);
```

```
    varA = 10;
```

```
varB = 10;
result = varA & varB;
Console.WriteLine("{0} AND {1} Result : {2}", varA, varB, result);
Console.ReadLine();
}
```

a) 0, 20  
b) 10, 10  
c) 0, 10  
d) 0, 0

**Đáp án: c**

**Giải thích:** Khi các phép toán 'OR' được thực hiện trên các giá trị nhị phân sau đây là kết quả của OR.

'OR' có nghĩa là phép toán cộng (+).

#### 84. Đầu ra của đoạn mã C # sau đây sẽ là gì?

```
public static void Main()
{
    byte varA = 10;
    byte varB = 20;
    long result = varA | varB;
    Console.WriteLine("{0} OR {1} Result :{2}", varA, varB, result);
    varA = 10;
    varB = 10;
    result = varA | varB;
    Console.WriteLine("{0} OR {1} Result : {2}", varA, varB, result);
}
```

a) 20, 10  
b) 30, 10  
c) 10, 20  
d) 10, 10

**Đáp án: b**

**Giải thích:** Có hai loại thao tác Shift là "Shift phải" và "Shift trái". Thao tác Right Shift được sử dụng để chuyển vị trí bit về phía bên phải. Thao tác Left Shift được sử dụng để chuyển vị trí bit về phía bên trái. Khi các thao tác Right Shift được thực hiện trên một giá trị nhị phân, các bit được dịch chuyển một vị trí về phía bên phải.

#### 85. Đầu ra của đoạn mã C # sau đây sẽ là gì?

```
static void Main(string[] args)
{
    byte b1 = 0 * AB;
    byte b2 = 0 * 99;
    byte temp;
    temp = (byte) ~b2;
    Console.Write( temp + " ");
    temp = (byte) (b1 << b2);
    Console.Write(temp + " ");
    temp = (byte)(b2 >> 2);
    Console.WriteLine(temp);
    Console.ReadLine();
}
```

a) 101 0 34

- b) 103 2 38
- c) 102 0 38
- d) 101 1 35

**Đáp án: c**

**86. Tùy chọn nào sau đây không phải là Toán tử Bitwise trong C#?**

- a) &, |
- b) ^, ~
- c) <<, >>
- d) +, -, =**

**Đáp án: d**

Giải thích: +, -, = là các toán tử gán trong C #.

**87. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
bool a = true;
bool b = false;
a |= b;
Console.WriteLine(a);
Console.ReadLine();
```

- a) 0
- b) 1
- c) True
- d) False

**Đáp án: c**

Giải thích: 'bools' là các bit đơn, vì vậy OR theo bit không ngoan giống với OR logic.

**88. Chọn bộ mã C # có liên quan để điền vào chỗ trống cho chương trình C # sau?**

```
static void Main(string[] args)
{
    int x = 10, y = 20;
    int res;
    /* _____ */
    Console.WriteLine(res);
}
```

- a) `x % y == 0 ? (x == y ? (x += 2):(y = x + y)):y = y*10;`
- b) `x % y == 0 ? y += 10:(x += 10);`
- c) `x % y == 0 ? return(x) : return (y);`
- d) All of the mentioned

**Đáp án: b**

**89. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    int y = 5;
    int x;
    int k = (!(Convert.ToInt32(y) > 10))? x = y + 3 : x = y + 10;
    Console.WriteLine(x);
    Console.WriteLine(y);
    Console.ReadLine();
}
```

- a) 5, 8
- b) 10, 4
- c) 8, 5

d) 11, 8

**Đáp án: c**

Giải thích: Vì điều kiện  $y > 10$  là false và  $!(\text{False}) = \text{true}$ . Vì vậy, câu lệnh đầu tiên  $x = y + 3$  được thực hiện là  $x = 8$  với  $y = 5$ .

### 90. Toán tử nào sau đây là toán tử điều kiện?

a) ':'

b) ?;

c) ?:

d) ??

**Đáp án: c**

Giải thích: Theo định nghĩa.

### 91. Đầu ra của đoạn mã C # sau đây sẽ là gì?

```
public static void Main(string[] args)
{
    int a = 4;
    int c = 2;
    bool b = (a % c == 0 ? true : false);
    Console.WriteLine(b.ToString());
    if (a/c == 2)
    {
        Console.WriteLine("true");
    }
    else
    {
        Console.WriteLine("false");
    }
    Console.ReadLine();
}
```

a) True False

b) False True

c) True True

d) False False

**Đáp án: c**

Giải thích:  $a \% c == 0$  điều kiện đúng là  $(4 \% 2 == 0)$ . Vì vậy, b được đánh giá là đúng. Bây giờ  $(a / c == 2)$  có nghĩa là nếu điều kiện cũng đúng, do đó nó được đánh giá là đúng.

### 92. Sắp xếp các toán tử theo thứ tự tăng dần như đã định nghĩa trong C #.

!, %, ?, :, &, ++, &&

a) ? : <&& <! = <& <++

b) ? : <&& <! = <++ <&

c) ? : <&& <& <! = <++

d) ? : <&& <! = <& <++

**Đáp án: c**

Giải thích: Theo định nghĩa.

### IF Statements

### 93. What will be the output of the following C# code?

```
static void Main(string[] args)
{
    int i = 30;
    int j = 25 % 25;
```

```

    if (Convert.ToBoolean(Convert.ToInt32(i = j)))
    {
        Console.WriteLine("In if");
    }
    else
    {
        Console.WriteLine("In else");
    }
    Console.WriteLine("In main");
    Console.ReadLine();
}

```

- a) In if
- b) In else
- c)
- In if
- In main
- d)**
- In else
- In main

[View Answer](#)

Answer: d

Explanation: Usage of '=' operator instead of '==' operator. Hence, the condition is not true.

Output:

**In else**

**In main**

**94. What will be the output of the following C# code?**

```

static void Main(string[] args)
{
    int i;
    int b = 8, a = 32;
    for (i = 0; i <= 10; i++)
    {
        if ((a / b * 2) == 2)
        {
            Console.WriteLine(i + " ");
            continue;
        }
        else if (i != 4)
            Console.Write(i + " ");
        else
            break;
    }
    Console.ReadLine();
}

```

- a) 1 2 3 4 5 6 7 8 9
- b) 0 1 2 3 4 5 6 7 8
- c) 0 1 2 3
- d) 0 1 2 3 4

[View Answer](#)

Answer: c

Explanation: The if condition will never be fulfilled as  $((a / b) * 2 == 2)$  is never true. Hence, only else part of condition will be executed until  $i \neq 4$  i.e  $i = 0, 1, 2, 3$ .

Output: **0 1 2 3**



**95. Which of the following conditions are true in the following C# code?**

```
static void Main(string[] args)
{
    Console.WriteLine("Enter a letter:");
    char c = (char)Console.Read();
    if (Char.IsDigit(c) == true)
        Console.WriteLine("A number");
    else if (char.IsLower(c) == true)
        Console.WriteLine("A lowercase letter");
    else if (char.IsUpper(c) == true)
        Console.WriteLine("An uppercase letter");
    Console.ReadLine();
}
```

- a. Enter a letter :  
a  
An uppercase letter
- b. Enter a letter :  
A  
An uppercase letter
- c. Enter a letter :  
2  
A number
- d. Enter a letter :  
2  
A lowercase letter.

- a) a, b, c  
b) b, c, d  
c) a, d, b  
d) b, c

[View Answer](#)

Answer: d

Explanation: None.

Output:

```
Enter a letter :
A
An uppercase letter
Enter a letter :
2
A number
```

**96. What will be the output of the following C# code?**

```
static void Main(string[] args)
{
    int i, j;
    for (i = 2; i >= 0; i--)
    {
        for (j = 0; j <= 2; j++)
        {
            if (i == j)
            {
                Console.WriteLine("1");
            }
            else
            {
                Console.WriteLine("0");
            }
        }
    }
}
```

```

        Console.WriteLine("0");
    }
}
Console.WriteLine("\n");
Console.ReadLine();
}
}

```

|       |       |       |       |
|-------|-------|-------|-------|
| a)    | b)    | c)    | d)    |
| 1 0 0 | 0 1 0 | 0 0 1 | 1 0 0 |
| 0 1 0 | 1 0 0 | 0 1 0 | 0 0 1 |
| 0 0 1 | 0 0 1 | 1 0 0 | 0 1 0 |

[View Answer](#)

Answer: c

Explanation: In first row for  $i = 2 : j = 0 == 0$  as if condition fails for  $(i == j)$

$i = 2 : j = 1 == 0$  as again if condition fails for  $(i == j)$

$i = 2 : j = 2 == 1$  as  $(i == j)$ .

In Second row for  $i = 1 : j = 0 == 0$  as if condition fails for  $(i == j)$

$i = 1 : j = 1 == 1$  (as  $i == j$ )

$i = 1 : j = 2 == 0$  as  $(i == j)$  not true

In Third row for  $i = 0 : j = 0 == 1$  as  $(i == j)$  true

$i = 0 : j = 1 == 0$  as  $(i == j)$  not true.

$i = 0 : j = 2 == 0$ .

So, 0 0 1

0 1 0

1 0 0

Output: 0 0 1

0 1 0

1 0 0

**97. Select the correct 'if statement' to be filled in the following C# code?**

```

static void Main(string[] args)
{
    int []num = {50, 65, 56, 88, 43, 52};
    int even = 0, odd = 0;
    for (int i = 0 ; i < num.Length ; i++)
    {
        /* _____ */
    }
    Console.WriteLine("Even Numbers:" +even);
    Console.WriteLine("Odd Numbers:" +odd);
    Console.ReadLine();
}

```

a)

```

if ((num % 2) == 0)
{
    even += 1;
}
else
{
    odd += 1;
}

```

b)

```

if((num * i) == 0)

```

```
{  
    even += 1;  
}  
else  
{  
    odd += 1;  
}
```

c)

```
if(num[i] % 2 == 0)  
{  
    even += 1;  
}  
else  
{  
    odd += 1;  
}
```

d)

```
if(num[i] % 2 = 0)  
{  
    even += 1;  
}  
else  
{  
    odd += 1;  
}
```

[View Answer](#)

Answer: c

Explanation:

```
int []num = {50, 65, 56, 88, 43, 52};  
int even = 0, odd = 0;  
for (int i = 0 ; i < num.Length ; i++)  
{  
    if (num[i] % 2 == 0)  
    {  
        even += 1;  
    }  
    else  
    {  
        odd += 1;  
    }  
}  
Console.WriteLine("Even Numbers: " + even);  
Console.WriteLine("Odd Numbers: " + odd);  
Console.ReadLine();
```

**98. What will be the output of the following C# code?**

```
static void Main(string[] args)  
{  
    int a = 15, b = 10, c = 1;  
    if (Convert.ToBoolean(a) && (b > c))  
    {  
        Console.WriteLine("cquestionbank");  
    }  
    else
```

```
{  
    break;  
}
```

- a) cquestionbank
- b) It will print nothing
- c) Compile time error
- d) Run time error

[View Answer](#)

Answer: c

Explanation: Keyword "break" is not part of if-else statement. This keyword is used in case of loop or switch case statement.

**99. What will be the output of the following C# code?**

```
static void Main(string[] args)  
{  
    int a = 5;  
    if (Convert.ToBoolean((.002f) -(0.1f)))  
        Console.WriteLine("Sachin Tendulkar");  
    else if (a == 5)  
        Console.WriteLine("Rahul Dravid");  
    else  
        Console.WriteLine("Ms Dhoni");  
    Console.ReadLine();  
}
```

- a) Rahul Dravid
- b) Sachin Tendulkar
- c) Ms Dhoni
- d) Warning : Unreachable Code

[View Answer](#)

Answer: b

Explanation:  $(0.002 - 0.1f)$  not equivalent to zero hence it is true. So, only first if clause will execute and print: Sachin Tendulkar on console. As, first condition is always true so no else if statement will be executed.

Output:

Sachin Tendulkar

**100. What will be the output of the following C# code?**

```
1. static void Main(string[] args)  
2. {  
3.     int a = -1;  
4.     int b = -1;  
5.     if (Convert.ToBoolean(++a == ++b))  
6.         Console.WriteLine("a");  
7.     else  
8.         Console.WriteLine("b");  
9.     Console.ReadLine();  
10. }
```

- a) a
- b) b
- c) Compile time error
- d) Code execute successfully with no output

[View Answer](#)

Answer: c

Explanation: Both a and b are constants. Illegal to assign a value to constant on left hand of '=' operator. Hence, it must be some variable.

**101. What will be the output of the following C# code?**

```
static void Main(string[] args)
{
    int a = 5, b = 10;
    if (Convert.ToBoolean(Convert.ToInt32(0xB)))
    if (Convert.ToBoolean(Convert.ToInt32(022)))
    if (Convert.ToBoolean(Convert.ToInt32('\xeb')))
        Console.WriteLine("java");
    else ;
    else ;
    else ;
}
```

- a) Compile time error: Misplaced else
- b) Compile time error: Undefined symbol
- c) java
- d) Warning: Condition is always true

[View Answer](#)

Answer: c

Explanation:

0xB: hexadecimal integer constant.  
022: It octal integer constant.  
'\xeb': It is hexadecimal character constant.

As zero is false and any non-zero number is true. All, constants return a non-zero value. So, all if conditions in the above program are true.

Output:

java.

**102. What will be the output of the following C# code?**

```
static void Main(string[] args)
{
    int a = 5, b = 10;
    if (Convert.ToBoolean(Convert.ToInt32(++a)) ||
        Convert.ToBoolean(Convert.ToInt32(++b)))
    {
        Console.WriteLine(a + "\n" + b);
    }
    else
        Console.WriteLine(" C# ");
}
```

- a) 6 11
- b) 6 16
- c) 6 12
- d) 6 10

[View Answer](#)

Answer: d

Explanation: Consider the following expression: ( ++a || ++b). In this expression || is 'Logical OR operator'. Two important properties of this operator are:

Property 1:

(Expression1) || (Expression2)

|| operator returns 0 if and only if both expressions return a zero otherwise || operator returns 1.

initial value of a is 5. So ++a will be 6. Since ++a is returning a non-zero so ++b will not execute.

Output : 6 10.

### Switch Statements

#### 103. Đầu ra của đoạn mã C # sau đây sẽ là gì?

```
static void Main(string[] args)
{
    int movie = 1;
    switch (movie << 2 + movie)
    {
    default:
        Console.WriteLine("3 Idiots");
        break;
    case 4:
        Console.WriteLine("Ghazini");
        break;
    case 5:
        Console.WriteLine("Krishh");
        break;
    case 8:
        Console.WriteLine("Race");
        break;
    }
    Console.ReadLine();
}
```

a) 3 Idiots

b) Ghazini

**c) Race**

d) Krishh

Trả lời: c

Giải thích: Chúng ta có thể đặt trường hợp 'mặc định' theo bất kỳ thứ tự nào và do đó viết các trường hợp theo bất kỳ thứ tự nào.

#### 104. Đầu ra của đoạn mã C # sau đây sẽ là gì?

```
static void Main(string[] args)
{
    int i = 2, j = 4;
    switch (i + j * 2)
    {
    case 1 :
    case 2 :
        Console.WriteLine("1 and 2");
        break;
    case 3 to 10:
        Console.WriteLine("3 to 10");
        break;
    }
    Console.ReadLine();
}
```

a) 3 đến 10 sẽ được in

b) 1 và 2 sẽ được in

**c) Mã báo lỗi thiếu; trước :**

d) Mã cho đầu ra là 3 đến 10

**Trả lời: c**

Giải thích: Lỗi cú pháp- trường hợp chuyển đổi không hoạt động với cú pháp từ 3 đến 10:

Đầu ra:

Ở đây  $i = 2, j = 4$ . Vì vậy,  $(i + j * 2)$  cho đầu ra là 10 và trường hợp 10 bị thiếu.

Vì vậy, không in gì cho mã đã cho.

**105. Đầu ra của mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    int i = 2, k = 3;
    switch (i - k)
    {
        case -1:
            ++i;
            ++k;
            break;
        case 2:
            --i;
            ++k;
            break;
        default:
            i += 3;
            k += i;
            break;
    }
    Console.WriteLine(i + "\n" + k);
    Console.ReadLine();
}
```

a) 2 3 3

b) 3 2 3

**c) 3 4 4**

d) 5 10 10

**Trả lời: c**

Đầu ra:

3

4

4

Giải thích:  $i - k = -1$ . Vì vậy, trường hợp -1 sẽ chỉ được thực hiện.

**106. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    int const p = 0;
    switch (3 * 5 / 6)
    {
        case p:
            Console.WriteLine("A");
            break;
        case p * 1:
            Console.WriteLine("B");
            break;
        case p - 2:
```

```

        Console.WriteLine("C");
        break;
    default:
        Console.WriteLine("D");
    }
}

```

- a) A
- b) B
- c) C

**d) Lỗi thời gian biên dịch**

Trả lời: d

Giải thích: Trong biểu thức chữ hoa, chúng ta không có biến hằng số.

**107. Đầu ra của mã C # sau đây sẽ là gì?**

**static void Main(string[] args)**

```

{
    int i = 2, j = 3, k = 4;
    switch (i + j - k)
    {
        case 0: case 2: case 4:
            ++i;
            k += j;
            break;
        case 1: case 3: case 5 :
            --i;
            k -= j;
            break;
        default:
            i += j;
            break;
    }
    Console.WriteLine(i + "\n" + j + "\n" + k);
    Console.ReadLine();
}

```

**a) 1 3 1**

- b) 2 3 4
- c) 5 3 4

d) Lỗi thời gian biên dịch

**Trả lời: a**

Giải thích: Giải biểu thức  $(i + j - k)$  cho 1 và do đó, giải cho trường hợp 1: trường hợp 3: trường hợp 5:.

Đầu ra: 1 3 1

**108. Đầu ra của mã C # sau đây sẽ là gì?**

**static void Main(string[] args)**

```

{
    int i = 9, j = 7;
    switch (i - j + 3)
    {
        case 9: 7:
            j += 6;
            break;
        case 5:

```



```
        i -= 4;
        break;
    }
    Console.WriteLine(i + "\n" + j);
    Console.ReadLine();
}
```

a) 5 7

b) 9 13

**c) Lỗi thời gian biên dịch**

d) 9 7

**Trả lời: c**

Giải thích: Biểu thức '7: 'không hợp lệ trong trường hợp 9: 7:

**109. Đầu ra của mã C # sau đây sẽ là gì?****static void Main(string[] args)**

```
{
    switch (5)
    {
        case 5.0f:
            Console.WriteLine("harsh");
            break;
        case 5:
            Console.WriteLine("amish");
            break;
        case 5.0L:
            Console.WriteLine("ANKIT");
            break;
        default:
            Console.WriteLine("ashish");
    }
    Console.ReadLine();
}
```

a) amish

b) ANKIT

c) harsh

**d) Lỗi thời gian biên dịch.**

Trả lời: d

Giải thích: Chỉ cho phép các giá trị tích phân cho biểu thức chữ hoa chữ thường.

5,0f = (int) 5,0f.

5.0L = (int) 5.0L.

**110. Đầu ra của mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    int i;
    int j = 1;
    int []ar = {21, 22, 13, 4};
    switch (ar[j])
    {
        case 1:
            i++;
    }
}
```

```

        break;
    case 2:
        i += 2;
        j = 3;
        continue;
    case 3:
        i %= 2;
        j = 4;
        continue;
    default:
        --i;
    }
    Console.WriteLine(i);
    Console.ReadLine();
}

```

a) 23

b) 15

**c) Lỗi thời gian biên dịch.**

d) 12

Trả lời: c

Giải thích: Không thể sử dụng Continue như một phần của câu lệnh switch.

**111. Đầu ra của mã C # sau đây sẽ là gì?****static void Main(string[] args)**

```

{
    char ch = Convert.ToChar('a' | 'b' | 'c');
    switch (ch)
    {
        case 'A':
        case 'a':
            Console.WriteLine("case A|case a");
            break;
        case 'B':
        case 'b':
            Console.WriteLine("case B|case b");
            break;
        case 'C':
        case 'c':
        case 'D':
        case 'd':
            Console.WriteLine("case D|case d");
            break;
    }
    Console.ReadLine();
}

```

a) Lỗi thời gian biên dịch.

b) case A|case a

c) case B|case b

**d) case D |case d**

Trả lời: d

Giải thích: Câu lệnh trường hợp được khai báo cuối cùng sẽ chỉ được thực thi vì không có số trường hợp cụ thể nào được khai báo sẽ được gọi.

Đầu ra:

trường hợp D | trường hợp d

**112. Đầu ra của mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    char ch = 'p';
    switch (ch)
    {
        case 'p':
            Console.WriteLine("coco" + "\t" + Convert.ToInt32(ch));
            break;
        default:
            Console.WriteLine("default");
            break;
    }
    Console.WriteLine("main");
}
```

a) coco main  
b) coco 112  
**c) coco 112 main**  
d) Lỗi thời gian biên dịch.

**Trả lời: c**

Giải thích: Giá trị ASCII của 'p' là 112. Do đó, coco 112 main.

Đầu ra: coco 112 main.

**For Loop Statements****113. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
1. static void Main(string[] args)
2. {
3.     int i;
4.     for (i = 0; ; )
5.     {
6.         Console.WriteLine("hello");
7.     }
8.     Console.ReadLine();
9. }
```

a) No output

b) hello

**c) hello printed infinite times**

d) Code will give error as expression syntax

[View Answer](#)

Answer: C

Explanation: Testing condition for the loop is absent. So, loop will continue executing.

Output :

```
hello
hello
hello
.
.
```

**114. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
1.     float f;
2.     for (f = 0.1f; f <= 0.5; f += 1)
3.     {
4.         Console.WriteLine(++f);
5.     }
```

```
5. Console.ReadLine();
6. }
```

**a) 1.1**

b) 0.1

c) 0.1 0.2 0.3 0.4 0.5

d) None of the mentioned

[View Answer](#)

Answer: A

Explanation:  $f = 0.1$  and  $++f = 0.1 + 1 = 1.1$ . So,  $1.1 > 0.5$ , Condition fails and hence loop terminates.

Output : **1.1****115. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    int I, X;
    for (I = 1; I <= (9 % 2 + I); I++)
    {
        X = (I * 3 + I * 2) / I;
        Console.WriteLine(X);
    }
    Console.ReadLine();
}
```

a) Output of code is 5 10

b) Output is 5 5 5 5

c) Print 5 infinite times

d) None of the mentioned

[View Answer](#)

Answer: C

Explanation: Testing condition is always incremented i.e  $i$  never  $>$  ( $9\%2+I$ ). So, loop will never terminate.

Output : **5 5 5.....****116 Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    int I, J = 0;
    for (I = 1; I < 10; ) ;
    {
        J = J + I;
        I += 2;
    }
    Console.WriteLine("Sum of first 10 even numbers is:" + J);
    Console.ReadLine();
}
```

a) 1 2 3 4 5 6 7 8 9

b) 25

c) 1

**d) Run time error**[View Answer](#)

Answer: D

Explanation: Due to presence of ``;` after `for()` loop condition do not work as according to the statement. Remove the ``;`.

Output : **25**

**117. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    int i = 5;
    for (; Convert.ToBoolean(Convert.ToInt32(i)); Console.WriteLine(i--)) ;
    Console.ReadLine();
}
```

a) 4 3 2 1

b) 3 2 1

**c) 5 4 3 2 1**

d) 2 1

[View Answer](#)

Answer: C

Explanation: Since, i = 5 and test condition is executed until i!=0. So, i-- decrements value of i till condition is satisfied.

Output: 5 4 3 2 1

**118. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    int i, s = 0;
    for (i = 1; i <= 10; s = s + i, i++);
    {
        Console.WriteLine(s);
    }
    Console.ReadLine();
}
```

a) Code report error

b) Code runs in infinite loop condition

c) Code gives output as 0 1 3 6 10 15 21 28 36 45

**d) Code give output as 55**[View Answer](#)

Answer: D

Explanation: Since occurrence of termination symbol(;) at end of for loop.

Output: 55

**119. Câu nào đúng trong số các câu đã nêu?**

i. The for loop works faster than a while loop

ii. for( ; ; ) implements an infinite loop

a) Only i is correct

**b) Only ii is correct**

c) Both i and ii are correct

d) Both i and ii are incorrect

[View Answer](#)

Answer: B

Explanation: By definition.

**120. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
{
    int i;
    Console.WriteLine("Hi");
    for (i = 1; i <= 10; i++)
        Program.Main(args);
    Console.ReadLine();
}
```

- a) Prints 'Hi' for one time
- b) Prints 'Hi' for infinite times
- c) Stack overflow exception Condition generated
- d) None of the mentioned

[View Answer](#)

Answer: C

Explanation: Occurrence of 'main()' condition after for loop.

Output:

```
Hi
Hi
```

stack overflow exception.

**121. Đoạn mã C # cần được thêm vào để có kết quả đầu ra sau đây?**

```
* * * * *
* * * *
* * *
* *
*
static void Main(string[] args)
{
    int i,j;
    /* Add code here */
}
```

a)

```
for (i = 0; i <= 4; i++)
{
    for(j = 0; j <= 4; j++)
        console.WriteLine("*");
    console.WriteLine("\n");
}
```

b)

```
for (i = 0; i <= 4; i++)
{
    for(j = 4; j <= i; j--)
        console.WriteLine("*");
    console.WriteLine("\n");
}
```

c)

```
for (i = 0; i <= 4; i++)
{
    for (j = i; j <= 4; j++)
        console.WriteLine("*");
    console.WriteLine("\n");
}
```

d)

```
for ( i = 0; i <= 4; i++)
{
    for (j = 0; j <= i; j++)
        console.WriteLine("*");
    console.WriteLine("\n");
}
```

[View Answer](#)

Answer: C

Explanation: Input in Console and run the code.

**122. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    int i;
    for (i = -3; i <= 3; i++)
    {
        switch (i)
        {
            case 0:
                Console.WriteLine("zero");
                break;
        }
        if (i > 0)
            Console.WriteLine("A");
        else if (i < 0)
            Console.WriteLine("B");
    }
    Console.ReadLine();
}
```

- a) B B zero A A A
- b) B zero A A A
- c) B B B zero A A A
- d) A A A zero B B B

[View Answer](#)

Answer: C

Explanation: for i = -3, -2, -1 statement executed as B. for i = 0, it is zero and for i = 1, 2, 3 again statement printed as A separately for each value of i.

Output: B B B zero A A A

**123. Điều nào sau đây không phải là vòng lặp vô hạn?**

- a) for( ;'0'; )
- b) for( ;'0'; )**
- c) for( ;'1'; )
- d) for( ;'1'; )

[View Answer](#)

Answer: B

Explanation: None.

**124 Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    int i, j;
    for (i = 1, j = i; i <= 3 && j >= 0; i++, j--)
    {
        if (i == j)
            continue;
        else
            Console.WriteLine(j);
    }
    Console.ReadLine();
}
```

- a) i = 0, j = 1;
- b) i = 1, j = 0;

**c) j = 0;**

d) None of the mentioned

[View Answer](#)

Answer: C

Explanation: Since for  $i = 1, j = 1$  and  $1 \leq 3$  also  $1 \geq 0$  we had  $i == j$ . But after  $i++$  and  $j--$ . The initial value of 'j' which is '0' as  $j--$  preferred other than value of 'j' in  $i = j$ .

Output: j = 0

**125. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    int i = -10;
    for ( ; Convert.ToBoolean(Convert.ToInt32(i)) ; Console.WriteLine(i++)) ;
    Console.ReadLine();
}
```

a) -9 -8 -7 -6 -5 -4 -3 -2 -1

b) -10 -9 -8 -7 -6 -5 -4 -3 -2

**c) -10 -9 -8 -7 -6 -5 -4 -3 -2 -1**

d) -8 -7 -6 -5 -4 -3 -2 -1

[View Answer](#)

Answer: C

Explanation: For first value of  $i = -10$ . Condition is executed until  $i != 0$ .

Output:

**-10 -9 -8 -7 -6 -5 -4 -3 -2 -1**

### While Loop Statements

**126. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main ( string [ ] args )
{
    int i, j;
    cho ( i = 1 ; i <= 3 ; i ++ )
    {
        j = 1 ;
        trong khi ( i % j == 2 )
        {
            j ++;
        }
        Bảng điều khiển . WriteLine ( i + " " + j );
    }
    Bảng điều khiển . ReadLine ( ) ;
}
```

A. 11 21 31

B. 1 12 13 1

C. 11 21 31

D. 1 1 2 1 3 1

**Đáp án: C**

Giải thích: Vì, điều kiện không bao giờ thỏa mãn với bất kỳ giá trị nào của  $i$  và  $j$  mà  $(i \% j == 2)$ . Do đó,  $j$  luôn là số gia tăng '1' và 'i' không đổi đối với  $i = 1, 2, 3$ .

Đầu ra: 11 21 31

**127. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main ( string [ ] args )
```



```
{
    float s = 0.1f;
    trong khi ( s <= 0.5f )
    {
        ++ s;
        Bảng điều khiển . WriteLine ( các );
    }
    Bảng điều khiển . ReadLine ( );
}
```

- a) 0,1
- b) 1,1
- c) 0,1 0,2 0,3 0,4 0,5
- d) Không có đầu ra

**Đáp án: B**

Giải thích: Đối với điều kiện while đầu tiên, hãy kiểm tra khi  $s = 0$ . Nếu điều đó đúng khi điều khiển đi vào bên trong vòng lặp  $++ s$  thì giá trị của  $s$  tăng lên 1 là  $1 + 0.1 = 1.1$ . Vì vậy, đối với điều kiện tiếp theo, vòng lặp while không thành công và do đó, in ra giá trị cuối cùng của  $s$  là 1.1.

**128. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main ( string [ ] args )
{
    int i;
    i = 0;
    trong khi ( i ++ < 5 )
    {
        Bảng điều khiển . WriteLine ( i );
    }
    Bảng điều khiển . WriteLine ( " \n " );
    i = 0;
    trong khi ( ++ i < 5 )
    {
        Bảng điều khiển . WriteLine ( i );
    }
    Bảng điều khiển . ReadLine ( );
}
```

- A. 1 2 3  
1 2 3 4 5
- B. 1 2 3  
1 2 3 4
- C. 1 2 3 4 5  
1 2 3 4
- D. 1 2 3 4 5  
1 2 3 4 5

**Đáp Án:C**

Giải thích: Trong khi  $(i ++ < 5)$  giá trị hiện tại của 'i' được kiểm tra đầu tiên và do đó in giá trị tăng dần sau đó. Vì vậy,  $i = 1, 2, 3, 4, 5$ . Nhưng trong khi  $(++ i < 5)$  giá trị hiện tại được tăng lên trước tiên và sau đó kiểm tra giá trị đó với điều kiện đã cho và do đó sau đó in ra giá trị đó. Vì vậy,  $i = 1, 2, 3, 4$ .

**129. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main ( string [ ] args )
{
    int x = 0 ;
    trong khi ( x < 20 )
    {
        trong khi ( x < 10 )
        {
            nếu ( x % 2 == 0 )
            {
                Bảng điều khiển . WriteLine ( x ) ;
            }
            x ++ ;
        }
    }
    Bảng điều khiển . ReadLine ( ) ;
}
```

- a) .1 2 3 4 5 6 7 8 9 10  
11 12 13 14 15 16 17 18 19 20
- b) 0 2 4 6 8 10 12 14 16 18 20
- c) 0 2 4 6 8
- d) 0 2 4 6 8 10

**Đáp án: C**

Giải thích: Điều kiện vòng lặp while bên trong kiểm tra số chẵn từ 0 đến 10 và do đó in ra số giữa phạm vi đã cho.

**130. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main ( string [ ] args )
{
    int x ;
    x = Chuyển đổi . ToInt32 ( Console . ReadLine ( ) ) ;
    int c = 1 ;
    trong khi ( c <= x )
    {
        nếu ( c % 2 == 0 )
        {
            Bảng điều khiển . WriteLine ( "Thực thi Trong khi" + c + " \t " + "thời gian"
);
        }
        c ++ ;
    }
    Bảng điều khiển . ReadLine ( ) ;
}
cho x = 8 .
```

- a)
  - Thực thi While lần 1
  - Thực thi While lần 3
  - Thực thi While lần 5
  - Thực thi While lần 7
- b)
  - Thực thi While lần 2

Thực thi While lần 4  
Thực thi While lần 6  
Thực thi trong lần 8

c)

Thực thi While lần 1  
Thực thi While lần 2  
Thực thi While lần 3  
Thực thi While lần 4  
Thực thi While lần 5  
Thực thi While lần 6  
Thực thi While lần 7

d)

Thực thi While lần 2  
Thực thi While lần 3  
Thực thi While lần 4  
Thực thi While lần 5

### Đáp án: B

Giải thích: Kiểm tra điều kiện nếu số chia hết cho 2 thì nó sẽ in ra số chẵn lần như đã cho với x = 8 như vậy, in từ 2 đến 8 lần Tương tự, đối với x = 5, thực hiện lần 2 và 4.

### 131. Đầu ra của đoạn mã C # sau đây sẽ là gì?

```
static void Main ( string [ ] args )  
{  
    int n, r ;  
    n = Chuyển đổi . ToInt32 ( Console . ReadLine ( ) ) ;  
    While ( n > 0 )  
    {  
        r = n % 10 ;  
        n = n / 10 ;  
        Bảng điều khiển . WriteLine ( + r ) ;  
    }  
    Bảng điều khiển . ReadLine ( ) ;  
}  
cho n = 5432 .
```

a) 3245

b) 2354

c) 2345

d) 5423

### Đáp án: C

Giải thích: Đảo ngược số bằng vòng lặp while.

### 132. Cú pháp đúng cho câu lệnh while là:

a)

```
1.   trong khi  
2.   {  
3.  
4.  
5.  
6.   } ( điều kiện ) ;
```

b)

```
1.   while ( điều kiện )  
2.   {  
3.
```

4.  
5. } ;
- c)
1. while ( điều kiện )  
2. {  
3.  
4.  
5. }
- d)
1. while ( điều kiện ) ;  
2. {  
3.  
4.  
5. }

**Đáp án: C**

Giải thích: Theo định nghĩa.

**133. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main ( string [ ] args )
{
    float i = 1 . 0f, j = 0 . 05f ;
    While ( i < 2 . 0f && j <= 2 . 0f )
    {
        Bảng điều khiển . WriteLine ( i ++ - ++ j ) ;
    }
    Bảng điều khiển . ReadLine ( ) ;
}
```

- a) 0,05f  
b) 1,50f  
c) -0,04999995f  
d) 1,50f

**Đáp án: C**

Giải thích: While (i = 1,0f và j = 0,05f). Chúng tôi, có điều kiện '&&' cho '1'. Vì vậy, điều khiển đi vào vòng lặp while. Vì, i = 1 và i ++ = thực hiện đầu tiên sau đó tăng dần. Vì vậy, đầu tiên với giá trị 'i' là 1,0f và ++ j = giá số đầu tiên và sau đó thực hiện chúng ta có j = 1,05f và hoạt động Kể từ (i ++ - ++ j) cho chúng ta một số dấu âm. Vì vậy, chúng tôi có thể gần lựa chọn của mình vào tùy chọn '-0.04999995f' một cách rõ ràng. Bây giờ, vì i = 2.0f nên vòng lặp bị ngắt.

**134. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main ( string [ ] args )
{
    int i = 0 ;
    While ( i <= 50 )
    {
        nếu ( i % 10 == 0 )
            tiếp tục ;
        khác
            phá vỡ ;
        i + = 10 ;
        Bảng điều khiển . WriteLine ( i % 10 ) ;
    }
}
```

- a) mã in đầu ra là 0 0 0 0 0
- b) mã in đầu ra là 10 20 30 40 50
- c) vòng lặp vô hạn nhưng không in ra bất kỳ thứ gì**
- d) Lỗi tạo mã

**Đáp án: C**

**135. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main ( string [ ] args )  
{  
    int i = 1 , j = 1 ;  
    While ( ++ i <= 10 )  
    {  
        j ++;  
    }  
    Bảng điều khiển . WriteLine ( i + " " + j ) ;  
    Bảng điều khiển . ReadLine ( ) ;  
}
```

- a) 12 11
- b) 10 11
- c) 11 10**
- d) 11 12

**Đáp án: C**

Giải thích: Với ++ i, giá trị tăng đầu tiên sau đó thực hiện như vậy, đối với ++ ii là 11 và j ++ là thực thi đầu tiên sau đó tăng dần. Vì vậy, j = 10.

**136. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main ( string [ ] args )  
{  
    int i = 1 ;  
    While ( i <= 1 )  
    {  
        if ( 'A' < 'a' )  
        {  
            Bảng điều khiển . WriteLine ( "Xin chào ..." ) ;  
        }  
        khác  
        {  
            Bảng điều khiển . WriteLine ( "Chào ..." ) ;  
        }  
        i ++;  
    }  
    Bảng điều khiển . ReadLine ( ) ;  
}
```

- a) Xin chào...
- b) Xin chào....
- c) Xin chào ... vô hạn thời gian
- d) Xin chào vô số lần

**Đáp án: B**

Giải thích: Giá trị ascii của 'A' là 65 và 'a' là 97. Vì vậy, rõ ràng 'A' < 'a'.

**137. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
1. static void Main ( string [ ] args )  
2. {  
3.     int i = 0 ;
```

```

4. while ( i ++ != 0 ) ;
5.   Bảng điều khiển . WriteLine ( i ) ;
6.   Bảng điều khiển . ReadLine ( ) ;
7. }

```

- a) -127 đến +127
- b) 0 đến 127
- c) 1
- d) Điều kiện lặp vô hạn

**Đáp án: C**

Giải thích:  $i++$  = thực hiện đầu tiên sau đó tăng dần là  $i = 0$ . Vì vậy,  $i++ \neq 0$ , sai rõ ràng là  $i = 0$ . Bây giờ, điều khiển đi vào bên trong vòng lặp với  $i = 1$ . Vì vậy, câu lệnh in ra  $i = 1$ .

### DO WHILE LOOP STATEMENTS

**138. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```

static void Main(string[] args)
{
    int i = 1, j = 2, k = 3;
    do
    {
        Console.WriteLine((Convert.ToBoolean(Convert.ToInt32(i++)))
            && (Convert.ToBoolean(Convert.ToInt32(++j))));
    } while (i <= 3);
    Console.ReadLine();
}

```

- a) 0 0 0
- b) True True True
- c) 1 1 1
- d) False False False

[View Answer](#)

Answer: b

Explanation:  $1 \text{ AND } 1 = \text{True}$ . Similarly, non zero number || non zero number = True.

Output:

**True True True**

**139. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```

static void Main(string[] args)
{
    float i = 1.0f, j = 0.05f;
    do
    {
        Console.WriteLine(i++ - ++j);
    } while (i < 2.0f && j <= 2.0f);
    Console.ReadLine();
}

```

- a) 0.05
- b) -0.05
- c) 0.95
- d) -0.04999995

[View Answer](#)

Answer: d

Explanation: None.

Output :

**-0.04999995**

**140. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    int i = 1, j = 5;
    do
    {
        Console.WriteLine(i = i++ * j);
    }while (i <= 10);
    Console.ReadLine();
}
```

a) 5 10 15 20 25 30 35 40 45 50

**b) 5 25**

c) 5 11 16 21 26 31 36 41 46 51

d) 5 30

[View Answer](#)

Answer: b

Explanation: For first step of loop  $i = 1$ . So,  $i++ * j = 1 * 5 = 5$ . For second step of loop  $i = 5$ ,  $j = 5$ . So,  $i++ * j = 25$ . As,  $i = 25$  hence,  $25 \geq 10$  loop condition breaks.

Output:

**5 25.****141. Đối với chương trình C # chưa hoàn chỉnh dưới đây, đoạn mã C # nào sẽ không tạo ra vòng lặp vô hạn?**

```
static void Main(string[] args)
{
    int i = 1234, j = 0;
    /*ADD CODE HERE */
    Console.WriteLine(j);
}
```

a)

**do**

{

 $j = j + (i \% 10);$ **}while ((i = i / 10) != 0);**

b)

**do**

{

 $j = j + (i \% 10);$ **}while ((i / 10) != 0);**

c)

**do**

{

 $j = j + (i \% 10);$ **}while ((i \% 10) != 0);**

d)

**do**

{

 $j = j + (i \% 10);$ **}while ((i/10 == 0) != 0);**[View Answer](#)

Answer: a

Explanation: None.

Output :

```
static void Main(string[] args)
```

```
{
    int i = 1234,j = 0;
    do
    {
        j = j +( i % 10);

    }while ((i = i / 10)!= 0);
    Console.WriteLine(j);
}
```

142. Đầu ra của đoạn mã C # sau đây sẽ là gì?

```
static void Main(string[] args)
{
    long x;
    x = Convert.ToInt32(Console.ReadLine());
    do
    {
        Console.WriteLine(x % 10);
    }while ((x = x / 10) != 0);
    Console.ReadLine();
}
enter x = 1234.
```

- a) number of digits present in x
- b) prints '1'
- c) prints reverse of x
- d) prints sum of digits of 'x'

[View Answer](#)

Answer: c

Explanation: Reverse of digits using while loop statements.

Output:

4321.

143. Đầu ra của đoạn mã C # sau đây sẽ là gì?

```
static void Main(string[] args)
{
    int i, s = 0, a = 1, d;
    i = Convert.ToInt32(Console.ReadLine());
    do
    {
        d = i % (2 * 4);
        s = s + d * a;
    }while ((Convert.ToInt32(i = i / (2 * 4))) != 0
    && (Convert.ToBoolean(Convert.ToInt32((a = (a * 10))))));
    Console.WriteLine(s);
    Console.ReadLine();
}
enter i = 342.
```

- a) It finds binary equivalent of i
- b) It finds octal equivalent of i
- c) It finds sum of digits of i
- d) It finds reverse of i

[View Answer](#)

Answer: b

Explanation: None.

Output :



```
i = 342.  
s = 526.
```

**144. Cú pháp chính xác của vòng lặp do while là gì?**

a)

```
do;  
{  
    statement;  
}while (condition);
```

b)

```
do(condition)  
{  
    statement;  
}while;
```

c)

```
do  
{  
    statement;  
}while (condition)
```

d)

```
do  
{  
    statement;  
}while (condition);
```

[View Answer](#)

Answer: d

Explanation: By definition

Output:

```
do  
{  
    statement;  
}while (condition);
```

**145. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)  
{  
    int x = 10;  
    do  
    {  
        Console.WriteLine( x++);  
    }  
    while(Convert.ToBoolean(5) && Convert.ToBoolean(4) && Convert.ToBoolean(3)  
    && Convert.ToBoolean(2) && Convert.ToBoolean(1) && Convert.ToBoolean(0));  
    Console.ReadLine();  
}
```

a) 13

b) 15

c) 11

d) 10

[View Answer](#)

Answer: d

Explanation: Here in do while condition '&amp;&amp;' i.e 'AND' operator return '0' i.e false. So, as condition is false so program comes out of the loop.

Output :

10.

**146. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    int x;
    for (x = 10; x <= 15; x++)
        while (Convert.ToBoolean(Convert.ToInt32(x)))
        {
            do
            {
                Console.WriteLine(1);
                if (Convert.ToBoolean(x >> 1))
                    continue;
            } while (Convert.ToBoolean(0));
            break;
        }
    Console.ReadLine();
}
```

a) 0 0 0....infinite times

b) 1 1 1....infinite times

**c) 1 1 1 1 1 1**

d) System outoflow exception error

[View Answer](#)

Answer: c

Explanation: The execution of for loop is done for six consecutive times.

Output :

**1 1 1 1 1 1****147. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    int x = 0;
    do
    {
        x++;
        if (x == 5)
        {
            x++;
            continue;
            break;
        }
        Console.WriteLine(x + " ");
    }
    } while (x < 10);
```

a) 1 2 3 4 5

b) 10

c) 5 6 7 8 9 10

**d) 1 2 3 4 5 6 7 8 9 10**[View Answer](#)

Answer: d

Explanation: The condition will print the numbers from 1 to 10 when x == 5 and when x does not satisfy if condition until x &lt; 10.

Output:

**1 2 3 4 5 6 7 8 9 10****148. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    int x;
    for (x = 1; x <= 3; x++)
    {
        int j = 1;
        do
        {
            j++;
        } while (x % j == 2);
        Console.WriteLine(x + " " + j);
    }
    Console.ReadLine();
}
```

- a) 1 12 1 3 1
- b) 1 12 13 1
- c) 12 22 32
- d) 11 21 31

[View Answer](#)

Answer: c

Explanation: None.

Output :

### Continue, Goto Statements

**149. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    int i;
    Console.WriteLine("enter value of i:");
    i = Convert.ToInt32(Console.ReadLine());
    if (i < 7)
    {
        i++;
        continue;
    }
    Console.WriteLine("final value of i:" + i);
    Console.ReadLine();
}
```

- a) 12
- b) 11
- c) Lỗi thời gian biên dịch
- d) 13

*Xem câu trả lời*

Đáp án: c

Giải thích: Không thể sử dụng vòng lặp 'Tiếp tục' trong vòng lặp 'nếu'. thay while bằng if (i < 7).

Đầu ra: Lỗi thời gian biên dịch.

**150. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    int i;
    Console.WriteLine("enter value of i:");
    i = Convert.ToInt32(Console.ReadLine());
}
```

```

if ( i % 2 == 0)
    goto even;
else
{
    Console.WriteLine("number is odd:");
    Console.ReadLine();
}
even:
Console.WriteLine("number is even:");
Console.ReadLine();
}
for i = 4.

```

- a) số là số lẻ
- b) số là số chẵn
- c) Lỗi thời gian biên dịch
- d) không có điều nào được đề cập

*Xem câu trả lời*

Đáp án: c

Giải thích: "Nhãn không xác định 'thậm chí' trong hàm main (). Cú pháp 'goto Even;' không chính xác thay vì sử dụng 'goto Even;'.

**151. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```

static void Main(string[] args)
{
    int i = 1, j;
    do
    {
        for (j = 1; ; j++)
        {
            if (j > 2)
                break;
            if (i == j)
                continue;
            Console.WriteLine(i + " " + j);
        }
        i++;
    } while (i < 3);
    Console.ReadLine();
}

```

- a) 12 21
- b) 21 12
- c) 13 21
- d) 11 21

*Xem câu trả lời*

Đáp án: a

Giải thích: for i = 1. Khi điều khiển đi vào vòng lặp đầu tiên nếu điều kiện được kiểm tra đối với j = 1 và as (j > 2) là false. Điều khiển bây giờ được chuyển cho câu lệnh console với i = 1 và j = 2. Bây giờ, trong khi giá trị điều kiện của 'i' được phản ánh là 2 tức là i = 2 là i ++. Vì, điều khiển (i < 3) lại đi vào vòng lặp for với i = 2 nhưng j = 1 chứ không phải j = 2 đối với j ++ và do đó, một lần nữa điều kiện tương tự lại thực hiện cho câu lệnh console.

Đầu ra: 12 21

**152. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    int i = 10, j = 0;
label:
    i--;
    if (i > 0)
    {
        Console.WriteLine(i + " ");
        goto label;
    }
    Console.ReadLine();
}
```

- a) 1 2 3 4 5 6 7 8 9 10
- b) 10 9 8 7 6 5 4 3 2 1 0
- c) 9 8 7 6 5 4 3 2 1
- d) 10 9 8 7 6 5 4 3 2 1

*Xem câu trả lời*

Đáp án: c

Giải thích: cho  $i = 10$ , vòng lặp thực hiện lần đầu tiên trong vòng lặp 'if' là  $(i > 0)$  tức là  $(9 > 0)$  và do đó in ra '9'. Tương tự, điều kiện nhả được thực thi một lần nữa cho  $(i-)$  tức là  $(9-1 = 8)$  và do đó một lần nữa in ra  $i = 8$ . Theo cách này, điều kiện lặp thực thi như 9, 8 đến 3, 2, 1.

OUTPUT:

9 8 7 6 5 4 3 2 1

**153. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    int i = 0, j = 0;
    while (i < 2)
    {
        l1: i--;
        while (j < 2)
        {
            Console.WriteLine("hi\n");
            goto l1;
        }
    }
    Console.ReadLine();
}
```

- a) hi hi hi
- b) hi hi
- c) hi
- d) hi hi hi... ..infinite time

*Xem câu trả lời*

Đáp án: d

Giải thích: Vì,  $i-$  vì vậy, điều kiện kiểm tra cho 'i' không bao giờ thỏa mãn nên nó không thành công và do đó xảy ra vòng lặp vô hạn.

đầu ra:

hi hi hi .....

**154. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
1. static void Main(string[] args)
2. {
3.     int i = 0;
4.     if (i == 0)
5.     {
6.         goto label;
7.     }
8.     label: Console.WriteLine("HI...");
9.     Console.ReadLine();
10. }
```

- a) Xin chào... vô số lần
- b) Mã chạy không in gì
- c) Xin chào,
- d) Xin chào...

*Xem câu trả lời*

Đáp án: d

Giải thích: với  $i = 0$ , nếu điều kiện được thỏa mãn là  $(i == 0)$ . Vì vậy, câu lệnh nhả được in ra.

Đầu ra:

Chào

**155. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    int i = 0, j = 0;
l1: while (i < 2)
{
    i++;
    while (j < 3)
    {
        Console.WriteLine("loop\n");
        goto l1;
    }
}
Console.ReadLine();
}
```

- a) vòng lặp được in số lần vô hạn
- b) vòng lặp
- c) vòng lặp lặp
- d) Lỗi thời gian biên dịch

*Xem câu trả lời*

Đáp án: c

Giải thích: Vì vòng lặp while bên ngoài tức là  $\text{while } (i < 2)$  chỉ thực thi hai lần. Do đó, vòng lặp trong khi thực hiện lần thứ ba đối với  $(j < 3)$  không thể thỏa mãn điều kiện  $i < 2$  vì  $i = 2$ . Do đó, ngắt vòng lặp và điều khiển không hoạt động trong vòng lặp.

Đầu ra:

vòng lặp vòng lặp .

**156. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
1. static void Main(string[] args)
2. {
3.     int i = 0, k;
```

```
4.     label: Console.WriteLine(i);
5.     if (i == 0)
6.         goto label;
7.     Console.ReadLine();
8. }
```

a) 0 0 0 0

b) 0 0 0

c) 0 lần vô hạn

d) 0

*Xem câu trả lời*

Đáp án: c

Giải thích: Vì, điều kiện if luôn đúng. Vòng lặp sẽ tiếp tục thực hiện luôn mà không có bất kỳ điều kiện kết thúc nào.

Đầu ra: 0 0 0 ...

**157. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    int i = 0;
    int j = 0;
    for (i = 0; i < 4; i++)
    {
        for (j = 0; j < 3; j++)
        {
            if (i > 1)
                continue;
            Console.WriteLine("Hi \n");
        }
    }
    Console.ReadLine();
}
```

a) In hi 4 lần

b) In hi 3 lần

c) In hi 6 lần

d) In hi vô hạn

*Xem câu trả lời*

Đáp án: c

Giải thích: Không có.

Đầu ra:

```
Chào
Chào
Chào
Chào
Chào
Chào
```

**158. Chọn đầu ra cho bộ mã sau:**

```
static void Main(string[] args)
{
    int a = 0;
    int i = 0;
    int b;
    for (i = 0; i < 5; i++)
    {
        a++;
    }
}
```

```
        Console.WriteLine("Hello \n");  
        continue;  
    }  
    Console.ReadLine();  
}
```

- a) in hello 4 lần
- b) in hello 3 lần
- c) in hello 5 lần
- d) in hello vô hạn

*Xem câu trả lời*

Đáp án: c

Giải thích: Điều kiện thực hiện cho đến khi và trừ khi  $i < 5$ . Vì vậy, nó in "hello" cho đến khi điều kiện "i" được thỏa mãn.

Đầu ra:

```
xin chào  
xin chào  
xin chào  
xin chào  
xin chào
```

**159. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)  
{  
    Console.WriteLine("HI");  
    continue;  
    Console.WriteLine("Hello");  
    Console.ReadLine();  
}
```

- a) Xin chào, xin chào
- b) Xin chào
- c) Xin chào
- d) Lỗi thời gian biên dịch

*Xem câu trả lời*

Đáp án: d

Giải thích: Sự vắng mặt của bất kỳ điều kiện lặp nào để đưa ra quyết định ngắt hay tiếp tục.

### Fundamentals of Class

**160. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
class sample  
{  
    public int i;  
    public int[] arr = new int[10];  
    public void fun(int i, int val)  
    {  
        arr[i] = val;  
    }  
}  
class Program  
{  
    static void Main(string[] args)  
    {  
        sample s = new sample();  
        s.i = 10;  
        sample.fun(1, 5);  
    }  
}
```



```
s.fun(1, 5);  
Console.ReadLine();  
}  
}
```

a) sample.fun (1, 5) sẽ không hoạt động chính xác

b) si = 10 không thể hoạt động vì tôi là 'công khai'

c) sample.fun (1, 5) sẽ đặt giá trị là 5 trong arr [1]

d) s.fun (1, 5) sẽ hoạt động chính xác

Xem câu trả lời

Trả lời: a

Giải thích: Cần có tham chiếu Đối tượng cho trường, phương thức hoặc thuộc tính không tĩnh.

```
sample s = new sample();  
s.i = 10;  
sample.fun(1, 5);  
sample.fun(1, 5);  
Console.ReadLine();
```

**161. Điều nào sau đây được sử dụng để xác định thành viên của một lớp bên ngoài?**

a):

b) ::

c) #

d) không có cái nào được đề cập

Xem câu trả lời

Đáp án: b

Giải thích: Theo định nghĩa.

**162. Toán tử dùng để truy cập hàm thành viên của một lớp?**

a):

b) ::

c).

d) #

Xem câu trả lời

Đáp án: c

Giải thích: objectname. function name (các đối số thực tế);

**163. Khai báo lớp được chỉ định nhiều nhất là gì?**

a) type

b) scope

c) type & scope

d) không loại nào được đề cập.

Xem câu trả lời

Đáp án: c

Giải thích: Dạng khai báo chung của lớp trong C # là:

```
class class_name  
{  
    member variables  
    variable1;  
    variable2;  
    variableN;  
    method1(parameter_list)  
    {  
        method body  
    }  
    method2(parameter_list)  
    {  
        method body  
    }  
}
```

```

    }
    methodN(parameter_list)
    {
        method body
    }
}

```

**164. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```

class sample
{
    public int i;
    public int j;
    public void fun(int i, int j)
    {
        this.i = i;
        this.j = j;
    }
}
class Program
{
    static void Main(string[] args)
    {
        sample s = new sample();
        s.i = 1;
        s.j = 2;
        s.fun(s.i, s.j);
        Console.WriteLine(s.i + " " + s.j);
        Console.ReadLine();
    }
}

```

- a) Lỗi khi gọi s.fun () do mức không thể truy cập được
- b) Lỗi vì tham chiếu 'this' sẽ không thể gọi 'i' và 'j'

**c) 1 2**

- d) Chạy thành công nhưng không in gì

Xem câu trả lời

Đáp án: c

Giải thích: Biến 'i' và 'j' được khai báo với phạm vi public trong lớp mẫu được truy cập bằng cách sử dụng đối tượng của lớp 'mẫu' là 's'.

Đầu ra: 1 2

**165. Câu nào sau đây về các đối tượng trong "C #" là đúng?**

- a) Mọi thứ bạn sử dụng trong C # đều là một đối tượng, bao gồm Windows Forms và các điều khiển
- b) Các đối tượng có các phương thức và sự kiện cho phép chúng thực hiện các hành động
- c) Tất cả các đối tượng được tạo từ một lớp sẽ chiếm số byte bằng nhau trong bộ nhớ

**d) Tất cả**

Xem câu trả lời

Đáp án: d

Giải thích: Theo định nghĩa.

**166. "Một cơ chế liên kết mã và dữ liệu với nhau trong thao tác, đồng thời giữ an toàn cho cả hai bên ngoài sự can thiệp và sử dụng sai mục đích. Tóm lại, nó cách ly một mã và dữ liệu cụ thể khỏi tất cả các mã và dữ liệu khác. Một giao diện được xác định rõ ràng sẽ kiểm soát quyền truy cập vào mã và dữ liệu cụ thể đó. "**

- a) Tính trừu tượng
- b) Tính đa hình

c) Tính kế thừa

d) Tính đóng gói

Xem câu trả lời

Đáp án: d

Giải thích: Theo định nghĩa.

a) Abstraction

b) Polymorphism

c) Inheritance

d) Encapsulation

**167. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```

lớp z
{
    public int X ;
    public int Y ;
    public const int c1 = 5 ;
    public const int c2 = c1 * 25 ;
    public void set ( int a, int b )
    {
        X = a ;
        Y = b ;
    }
}

chương trình lớp học
{
    static void Main ( string [ ] args )
    {
        zs = new z ( ) ;
        s . bậ ( 10 , 20 ) ;
        Bảng điều khiển . WriteLine ( s . X + "" + s . Y ) ;
        Bảng điều khiển . WriteLine ( z . C1 + "" + z . C2 ) ;
        Bảng điều khiển . ReadLine ( ) ;
    }
}

```

|                     |                     |                      |                      |
|---------------------|---------------------|----------------------|----------------------|
| a)<br>10 20<br>5 25 | b)<br>20 10<br>25 5 | c)<br>10 20<br>5 125 | d)<br>20 10<br>125 5 |
|---------------------|---------------------|----------------------|----------------------|

Xem câu trả lời

Đáp án: c

Giải thích: Hàm thành viên () 'set' được truy cập bằng cách sử dụng đối tượng của lớp 'z' các giá trị được truyền dưới dạng tham số cho 'a' và 'b'. Vì, biến 'c1' và 'c2' là thành viên dữ liệu công khai của lớp 'z'. Chúng được truy cập bằng tên lớp.

Đầu ra:

```

10 20
5 125

```

**168. Cách khai báo đúng đối tượng của lớp sau là?**

class name

a) name n = new name();

b) n = name();

c) name n = name();

d) n = new name();

Xem câu trả lời

Answer: a

Giải thích: Không có.

**169. Các thành viên dữ liệu của một lớp theo mặc định là?**

- a) protected, public
- b) private, public
- c) private
- d) public

Xem câu trả lời

Đáp án: c

Giải thích: Không có.

**170. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
class z
{
    public int X;
    public int Y;
    public const int c1 = 5;
    public const int c2 = c1 * 25;
    public void set(int a, int b)
    {
        X = a;
        Y = b;
    }
}

class Program
{
    static void Main(string[] args)
    {
        z s = new z();
        s.set(10, 20);
        Console.WriteLine(s.X + " " + s.Y);
        Console.WriteLine(z.c1 + " " + z.c2);
        Console.ReadLine();
    }
}
```

- a) Syntax error
- b) {0} is in city{1} harsh new delhi
- c) harsh is in new delhi
- d) Run successfully prints nothing

Xem câu trả lời

Đáp án: c

Giải thích: Hàm thành viên show () được truy cập bằng cách sử dụng đối tượng của lớp 'z' là 'n' làm object.member ().

Đầu ra:

khắc nghiệt là ở delhi mới

**171. Đoạn mã C # sau đây ngụ ý điều gì?**

```
csharp abc ;
abc = new csharp ( ) ;
```

- a) Tạo đối tượng trên lớp csharp
- b) Tạo đối tượng kiểu csharp trên heap hoặc trên ngăn xếp tùy thuộc vào kích thước của đối tượng
- c) tạo tham chiếu c trên csharp và đối tượng kiểu csharp trên heap

d) tạo đối tượng kiểu csharp trên chõng

Xem câu trả lời

Đáp án: c

Giải thích: Không có.

**172. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
class test
{
    public void print()
    {
        Console.WriteLine("Csharp:");
    }
}
class Program
{
    static void Main(string[] args)
    {
        test t;
        t.print();
        Console.ReadLine();
    }
}
```

a) Mã chạy thành công không in ra gì

b) Mã chạy và in "Csharp"

c) Lỗi cú pháp vì t là biến chưa được gán không bao giờ được sử dụng

d) Không có biến nào được đề cập

Xem câu trả lời

Đáp án: c

Giải thích: đối tượng của class test nên được khai báo là test t = new test ();

```
test t = new test();
t.print();
Console.ReadLine();
```

### Reference Variables and Assignment

**173. Công cụ sửa đổi tham chiếu nào được sử dụng để xác định biến tham chiếu?**

a) &

b) ref

c) #

d) \$

Đáp án: b

Giải thích: Không có.

**174. Đầu ra của mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    int a = 5;
    fun1 (ref a);
    Console.WriteLine(a);
    Console.ReadLine();
}
static void fun1(ref int a)
{
    a = a * a;
}
```

- a) 5
- b) 0
- c) 20
- d) 25

Trả lời: d

Giải thích: Ở đây 'a' = 5. Bản sao của biến được truyền dưới dạng tham chiếu đến tham số 'a'.

Output: 25

**175. Đầu ra của mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    int[] arr = new int[] {1, 2, 3, 4, 5};
    fun1(ref arr);
    Console.ReadLine();
}

static void fun1(ref int[] array)
{
    for (int i = 0; i < array.Length; i++)
    {
        array[i] = array[i] + 5;
        Console.WriteLine(array[i] + " ");
    }
}
```

a) 6 7 8 9 10

b) 15 17 8 8 20

c) 15 17 8 29 20

d) Syntax error while passing reference of array variable

Trả lời: a

Giải thích: mảng 'arr' sau khi khai báo được truyền làm tham số tham chiếu.  $a[0] = 1 + 5 = 6$ .

$a[1] = 2 + 5 = 7$ .

.

.

$a[4] = 5 + 5 = 10$ .

Output : 15 17 8 29 20

**176. Đầu ra của mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    int a = 10 , b = 20;
    Console.WriteLine("Result before swap is: "+ a + " "+b);
    swap(ref a, ref b);
    Console.ReadLine();
}

static void swap(ref int i, ref int j)
{
    int t;
    t = i;
    i = j;
    j = t;
    Console.WriteLine("Result after swap is:"+ i + " "+j);
}
```

a)

Result before swap is: 20 10

Result after swap is: 20 10

b)

Result before swap is: 10 20

Result after swap is: 20 10

c)

Result before swap is: 10 20

Result after swap is: 10 20

d)

Result before swap is: 20 10

Result after swap is: 10 20

Đáp án: b

Giải thích: Sử dụng lệnh gọi theo tham số tham chiếu.

Output:

Result before swap is: 10 20.

Result after swap is: 20 10.

**177. Đầu ra của mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    int []a = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
    func(ref a);
    Console.ReadLine();
}
static void func(ref int[] x)
{
    Console.WriteLine(" numbers are:");
    for (int i = 0; i < x.Length; i++)
    {
        if (x[i] % 2 == 0)
        {
            x[i] = x[i] + 1;
            Console.WriteLine(x[i]);
        }
    }
}
```

a) numbers are : 2 4 6 8 10

b) numbers are : 3 5 7 9 11

c) numbers are : 2 3 4 5 6

d) none of the mentioned

Đáp án: b

Giải thích: Những số chia hết cho 2 là 2, 4, 6, 8, 10 và khi điều kiện của vòng lặp được thực hiện, nó sẽ tăng lên 1.

tức là đối với x [1] = 2% 2 == 0. Vì vậy, x [1] = 2 + 1 = 3.

x [3] = 4% 2 == 0. Vì vậy, x [3] = 4 + 1 = 5, v.v.

Output : 3 5 7 9 11

**178. Chọn phát biểu sai về từ khóa 'ref' trong C #?**

a) References can be called recursively

b) The 'ref' keyword causes arguments to be passed by reference

c) When 'ref' are used, any changes made to parameters in method will be reflected in variable when control is passed back to calling method

d) All of the mentioned

Trả lời: a

Giải thích: Không có.

**179. Chọn sự khác biệt chính xác giữa '=' và '==' trong C #)**

'==' operator is used to assign values from one variable to another variable

'=' operator is used to compare value between two variables

b)

'=' operator is used to assign values from one variable to another variable

'==' operator is used to compare value between two variables

c) No difference between both operators

d) None of the mentioned

Đáp án: b

Giải thích: Không có.

**180. Đầu ra của mã C # sau đây sẽ là gì?**

```
1. static void Main(string[] args)
2. {
3.     int X = 0;
4.     if (Convert.ToBoolean(X = 0))
5.         Console.WriteLine("It is zero");
6.     else
7.         Console.WriteLine("It is not zero");
8.     Console.ReadLine();
9. }
```

a) It is zero

b) It is not zero

c) Lặp vô hạn

d) Các câu trên đều sai.

Đáp án: b

Giải thích: Toán tử '=' được sử dụng không phải là toán tử so sánh mà là toán tử gán. Vì giá trị được gán cho 'X' = 0. Vì vậy, giá trị '0' được lưu trữ trong 'X' và với sự trợ giúp của việc triển khai điều kiện if, nó được chuyển đổi thành 'false' có nghĩa là nó không phải là 0 mà là '1' có nghĩa là 'thật'.

**181. Đầu ra của mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    int X = 6,Y = 2;
    X *= X / Y;
    Console.WriteLine(X);
    Console.ReadLine();
}
```

a) 12

b) 6

c) 18

d) Compile time error

Trả lời: c

Giải thích:  $X * = X / Y$ . $X = x * (X / Y)$ .

Output: 18

**182. Đầu ra của mã C # sau đây sẽ là gì??**

```
static void Main(string[] args)
{
    int x = 4 ,b = 2;
    x -= b/= x * b;
    Console.WriteLine(x + " " + b);
    Console.ReadLine();
}
```



a) 4 2

b) 0 4

c) 4 0

d) 2 2

Trả lời: c

Giải thích:  $x = x - b$  và  $b = b / (x * b)$ . Output: 4 0**183. Đầu ra của biểu thức C # sau đây sẽ là gì?**`int a+= (float) b/= (long)c.`

a) float

b) int

c) long

d) không phải các kết quả trên.

Đáp án: b

Giải thích: Không có.

**184. Đầu ra của mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    int x = 8;
    int b = 16;
    int C = 64;
    x /= b /= C;
    Console.WriteLine(x + " " + b + " " + C);
    Console.ReadLine();
}
```

a) 8 2 32

b) 32 4 8

c) 32 2 8

d) Compile time error

Trả lời: d

Giải thích: Lỗi xử lý ngoại lệ của phép chia cho không.

**185. Đầu ra của mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    int x = 8;
    int b = 16;
    int C = 64;
    x /= b /= C;
    Console.WriteLine(x + " " + b + " " + C);
    Console.ReadLine();
}
```

a) 8 2 32

b) 32 4 8

c) 32 2 8

d) Compile time error

Trả lời: c

Giải thích:  $x / = b / = C$  là  $x = x * c / b$ .

Output: 32 2 8

### Methods in Class Phương Thức Trong Lớp

**186. Đầu ra của mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
```

```
{
    int a = 5;
    int s = 0, c = 0;
    Mul (a, ref s, ref c);
    Console.WriteLine(s + "t " +c);
    Console.ReadLine();
}
static void Mul (int x, ref int ss, ref int cc)
{
    ss = x * x;
    cc = x * x * x;
}
```

a) 125 25

**b) 25 125**

c) Biên dịch lỗi

d) 0 0

**Câu trả lời: b**

Giải thích: Giá trị của biến a được truyền theo giá trị trong khi giá trị của biến s và c được chuyển bằng tham chiếu.

**187. Câu nào sau đây đúng về hàm?****a) C # cho phép một hàm có các đối số với các giá trị mặc định**

b) Việc xác định lại một tham số phương thức trong phần thân của phương thức gây ra một ngoại lệ

c) C # cho phép hàm có các đối số với giá trị mặc định

d) Bỏ qua kiểu trả về trong định nghĩa phương thức dẫn đến ngoại lệ

**Câu trả lời: a****188. Đầu ra của mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    Mul();
    m();
    Console.ReadLine();
}
static void Mul()
{
    Console.WriteLine("4");
}
static void m()
{
    Console.WriteLine("3");
    Mul();
}
```

a) 4 3 3

b) 4 4 3

**c) 4 3 4**

d) 3 4 4

**Câu trả lời: c**

Giải thích: Đầu tiên Mul () sẽ được thực thi để in số '4' sau đó hàm m () sẽ được thực hiện để in số '3' và ở lần cuối hàm Mul () sẽ được thực hiện để in câu lệnh 4 để trả về đầu ra là 4 3 4.

**189. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    m();
    Console.ReadLine();
}
static void m()
{
    Console.WriteLine("HI");
    m();
}
```

- a) HI HI HI  
 b) HI  
 c) Ngoại lệ tràn ngăn xếp  
 d) Lỗi thời gian biên dịch

Câu trả lời: c

Giải thích: Điều khiển câu lệnh khi nhập cho một lần trong m () không đi ra ngoài, sau đó nó thực thi lặp đi lặp lại bên trong khối cho đến khi ngoại lệ tràn ngăn xếp xảy ra.

**190. Khi một hàm fun () nhận một số nguyên, một đơn & một đôi và trả về một số thập phân, thì cách chính xác để xác định hàm C # này là?**

a)

```
static fun(int i, single j, double k)
{
    return decimal;
}
```

b)

```
static decimal fun(int i, single, double k)
{
}
```

c)

```
decimal fun(int i, single j, double k)
{
}
```

d)

```
decimal static fun(int i, single j, double k)
{
}
```

Câu trả lời: b

Giải thích: Cách khai báo đúng của hàm được định nghĩa là tên\_kiểu\_lệnh của hàm (kiểu trả về).

**191. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    int i = 10;
    double d = 35.78;
    fun(i);
    fun(d);
    Console.ReadLine();
}
static void fun(double d)
{
    Console.WriteLine(d);
}
```

- |    |    |    |                      |
|----|----|----|----------------------|
| a) | b) | c) | d) Không có điều nào |
|----|----|----|----------------------|

|             |             |             |             |
|-------------|-------------|-------------|-------------|
| 35.78<br>10 | 10<br>35.00 | 10<br>35.78 | được đề cập |
|-------------|-------------|-------------|-------------|

**Câu trả lời: c**

Giải thích: Kiểu dữ liệu 'int' là kiểu dữ liệu con của 'double'. Do đó, khi phần đầu tiên của func () được thực thi thì nó là phần nguyên và do đó khi phần thứ hai được thực thi thì nó là phần nhân đôi.

**192. Một hàm trả về bao nhiêu giá trị?**

a) 0

b) 2

**c) 1**

d) Bất kỳ số lượng giá trị nào

**Câu trả lời: c**

Giải thích: Một phương thức chỉ có thể trả về một giá trị duy nhất hoặc không có giá trị nào nếu không có thì nó được khai báo là void method ();

**193. Đầu ra của mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    int y = 3;
    y++;
    if (y <= 5)
    {
        Console.WriteLine("hi");
        Main(args);
    }
    Console.ReadLine();
}
```

a) hi hi

b) hi

**c) Ngoại lệ tràn ngăn xếp**

d) Không có điều nào được đề cập

**Câu trả lời: c**

Giải thích: Nếu vòng lặp không bao giờ kết thúc, nó sẽ thực thi liên tục. Điều khiển không bao giờ nằm ngoài câu lệnh 'if'.

**194. Câu lệnh trả về nào trả về kết quả đầu ra một cách chính xác?**

**a)**

```
public int cube(int x)
{
    return (x + x);
}
```

b)

```
public int cube(int x)
return (x + x);
```

c)

```
public int cube(int x)
{
    return x + x;
}
```

d) Không có đề cập nào

**Câu trả lời: a**

Giải thích: Cú pháp chính xác của câu lệnh return được định nghĩa trong khối câu lệnh là {return (câu lệnh);}.

**195. Đầu ra của mã C # sau đây sẽ là gì?**

```
public static void Main(string[] args)
{
    p();
    void p()
    {
        Console.WriteLine("hi");
    }
}
```

**a) Compile time error**

b) hi

c) hi vô số lần

d) Không có đề cập nào

**Câu trả lời: a**

Giải thích: Định nghĩa không hợp lệ của hàm p() bên trong hàm main ().

**CONSTRUCTORS IN CLASS****196. Số lượng constructor mà một lớp có thể định nghĩa là ?**

a) 1

b) 2

**c) Bất kỳ số nào**

d) Không có nội dung nào được đề cập

**Trả lời: c**

Giải thích: Constructor là một phương thức đơn giản có cùng tên với class và do đó được sử dụng để tạo object của class.C# class có thể định nghĩa bất kỳ số lượng constructor nào. Mỗi lớp chứa một constructor mặc định.

**197. Cách xác định chính xác constructor của class đã cho như và khi các đối tượng của class được tạo ra là:**

maths s1 = new maths();

maths s2 = new maths(5, 5.4f);

|                                                                                                                                   |                      |
|-----------------------------------------------------------------------------------------------------------------------------------|----------------------|
| <b>a)</b><br><pre>public maths(int pp, single tt) {     p = pp;     t = tt; }</pre>                                               | b) sample s;         |
| <b>c)</b><br><pre>public sample() {     p = 0;     t = 0.0f; } public sample(int pp, single tt) {     p = pp;     t = tt; }</pre> | d) s = new sample(); |

**Trả lời: a**

Giải thích: Không có.

**198. Cách xác định chính xác đối tượng của lớp mẫu trong đó mã C# sẽ hoạt động chính xác là:**

```
class abc
{
    int i;
    float k;
    public abc(int ii, float kk)
    {
        i = ii;
        k = kk;
    }
}
```

- a) `abc s1 = new abc(1);`
- b) `abc s1 = new abc();`
- c) `abc s2 = new abc(1.4f);`
- d) `abc s2 = new abc(1, 1.4f);`

Trả lời: d

Giải thích: Trả về các loại tham số của đối tượng khớp lớp với các loại đối số constructor được xác định.

**199. Tuyên bố chính xác về constructors trong C#.NET là?**

- a) Nhà thầu thi công có thể bị quá tải
- b) Nhà xây dựng không bao giờ được gọi rõ ràng
- c) Nhà thầu xây dựng có cùng tên với tên của lớp
- d) Tất cả các đề cập

Trả lời: d

Giải thích: Không có.

**200. Trong số những điều sau đây là tuyên bố chính xác: constructors được sử dụng để ?**

- a) khởi tạo các đối tượng
- b) xây dựng các thành viên dữ liệu
- c) khởi tạo các đối tượng & xây dựng các thành viên dữ liệu
- d) không ai trong số những người được đề cập

Trả lời: a

Giải thích: Một khi đối tượng được khai báo có nghĩa là, constructor cũng được khai báo theo mặc định.

**201. Phương pháp add() có thể bị quá tải theo những cách sau trong C# không?**

```
public int add() { }
public float add(){ }
```

- a) Đúng

b) Sai

Trả lời: b

Giải thích: C# cung cấp tính năng quá tải phương thức có nghĩa là các phương thức có cùng tên nhưng các loại và đối số khác nhau.

**202. Những tuyên bố nào sau đây là chính xác về các constructors trong C#.NET?**

- a) Một constructors không được khai báo là tư nhân
- b) Nhà thầu thi công không thể quá tải
- c) Một constructors có thể là một constructors tĩnh
- d) Không có nội dung nào

Trả lời: c

Giải thích: Hàm xây dựng tĩnh là một constructor có thể được gọi trước khi bất kỳ đối tượng nào của class được tạo ra hoặc bất kỳ phương thức tĩnh nào được gọi. Một constructor tĩnh được gọi ngầm bởi .net CLR.

**203. Đầu ra của mã C# sau đây sẽ là gì?**

```

class abc
{
    public static void a()
    {
        console.WriteLine("first method");
    }
    public void b()
    {
        a();
        console.WriteLine("second method");
    }
    public void b(int i)
    {
        console.WriteLine(i);
        b();
    }
}
class program
{
    static void main()
    {
        abc k = new abc();
        abc.a();
        k.b(20);
    }
}

```

|                                                            |                                                           |
|------------------------------------------------------------|-----------------------------------------------------------|
| a)<br>second method<br>20<br>second method<br>first method | b)<br>first method<br>20<br>first method<br>second method |
| c)<br>first method<br>20                                   | d)<br>second method<br>20<br>first method                 |

View Answer

**Answer: b**

Explanation: The class 'abc' first calls method() 'a' then object of class 'k' when calls method 'b'. First of all, the method 'a' will be executed and then executes the second statement.

Output :

```

first method
20
first method
second method

```

**204. Kiểu trả về của constructors là gì?**

- a) int
- b) float
- c) void
- d) không phải các loại type trên

**Trả lời: d**

Giải thích: Nhà xây dựng không có bất kỳ loại trả lại nào thậm chí không có khoảng trống có trong đó.

**205. Phương pháp nào có cùng tên với phương pháp của lớp?**

a) delete

b) class

c) constructor

d) không phương pháp nào trong số những phương pháp được đề cập

Trả lời: c

Giải thích: Theo định nghĩa.

### DESTRUCTORS IN CLASS

**206. Toán tử nào trong số các toán tử sau biểu thị toán tử hủy?**

a) ::

b):

c) ~

d) &

**207. Phương thức được gọi bởi các máy khách của một lớp để giải phóng rõ ràng bất kỳ tài nguyên nào như mạng, kết nối, tệp đang mở, v.v. Khi đối tượng không còn được yêu cầu nữa?**

a) Hoàn thiện()

b) Kết thúc()

c) Vứt bỏ ()

d) Đóng ()

Giải thích: Vứt bỏ () chỉ là phương thức được gọi bởi các máy khách của một lớp để giải phóng rõ ràng bất kỳ tài nguyên nào như kết nối mạng, tệp đang mở, v.v. Khi đối tượng không còn được yêu cầu. Do đó, Vứt bỏ () cung cấp cho người lập trình khả năng điều khiển lập trình như vậy.

**208. Đặt tên cho một phương thức trùng tên với tên của lớp và được sử dụng để hủy các đối tượng cũng được gọi tự động khi ứng dụng cuối cùng đang trong quá trình kết thúc.**

a) Khởi xây dựng

b) Kết thúc ()

c) Người phá hủy

d) Kết thúc

Giải thích: Định nghĩa hàm hủy.

**209. Mệnh đề nào sau đây đúng?**

a) Có một bộ thu gom rác cho mỗi chương trình đang chạy trong bộ nhớ

b) Có một bộ thu gom rác chung cho tất cả các chương trình

c) Rác thu thập một đối tượng đặt tất cả các tham chiếu đến nó là null

d) Cả hai Có một bộ thu gom rác chung cho tất cả các chương trình & Để thu gom rác một đối tượng, hãy đặt tất cả các tham chiếu đến nó là null

**210. Toán tử được sử dụng để giải phóng bộ nhớ khi bộ nhớ được cấp phát?**

a) mới

b) miễn phí

c) xóa

d) không có cái nào được đề cập

Giải thích: 'Mới' được sử dụng để cấp phát bộ nhớ trong các hàm tạo. Do đó, chúng ta nên sử dụng "delete" để giải phóng bộ nhớ đó.

**211. Chọn câu lệnh sai về hàm hủy trong C #?**

a) Một lớp chỉ có thể có một hàm hủy

b) Các trình hủy không thể được kế thừa hoặc quá tải

c) Phần tử hủy có thể có các công cụ sửa đổi hoặc tham số

d) Tất cả những điều đã đề cập



**212. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
class sample
{
    int i;
    double k;
    public sample (int ii, double kk)
    {
        i = ii;
        k = kk;
        double j = (i) + (k);
        Console.WriteLine(j);
    }
    ~sample()
    {
        double j = i - k;
        Console.WriteLine(j);
    }
}
class Program
{
    static void Main(string[] args)
    {
        sample s = new sample(8, 2.5);
        Console.ReadLine();
    }
}
```

- a) 0 0
- b) 10,5 0
- c) Lỗi thời gian biên dịch
- d) 10,5 5,5**

*Giải thích: Hàm tạo đầu tiên 'mẫu' được gọi và do đó hàm hủy '~ mẫu' được đánh giá.*

*Đầu ra:*

*10,5, 5,5*

**213. Kiểu trả về của hàm hủy là gì?**

- a) int
- b) vô hiệu
- c) số
- d) không có cái nào được đề cập**

*Giải thích: Bộ hủy không có bất kỳ kiểu trả về nào thậm chí không có giá trị.*

**214. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
class box
{
    public int volume;
    int width;
    int height;
    int length;
    public box ( int w, int h, int l)
    {
        width = w;
        height = h;
        length = l;
    }
}
```

```

public ~box()
{
    volume = width * length * height;
}
}
class Program
{
    static void Main(string[] args)
    {
        box b = new box(4, 5, 9);
        Console.WriteLine(b.volume);
        Console.ReadLine();
    }
}

```

a) 0

b) 180

c) Lỗi thời gian biên dịch

d) Không có điều nào được đề cập

*Giải thích: Chúng tôi không thể sử dụng bất kỳ loại hỗ trợ nào với hàm hủy.***215. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```

class box
{
    public int volume;
    int width;
    int height;
    int length;
    public box ( int w, int h, int l)
    {
        this.width = w;
        this.height = h;
        this.length = l;
    }
    ~ box()
    {
        volume = this.width * this.length * this.height;
        console.writeline(volume);
    }
}
class Program
{
    public static void Main(string[] args)
    {
        box b = new box(4, 5, 9);
        Console.ReadLine();
    }
}

```

a) 0

b) Mã thực thi thành công nhưng không in gì

c) Lỗi thời gian biên dịch

d) 180

**ARRAY AND INITIALIZATION**

**216. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
1. static void Main(string[] args)
2. {
3.     int i, j;
4.     int[,] arr = new int[ 3, 3];
5.     for (i = 0; i < 3; ++i)
6.     {
7.         for (j = 0; j < 3; ++j)
8.         {
9.             arr[i, j] = i * 2 + i * 2;
10.            Console.WriteLine(arr[i, j]);
11.        }
12.        Console.ReadLine();
13.    }
14. }
```

a) 0, 0, 0 4, 4, 4 8, 8, 8

b) 4, 4, 4 8, 8, 8 12, 12, 12

c) 8, 8, 8 12, 12, 12 16, 16, 16

d) 0, 0, 0 1, 1, 1 2, 2, 2

[View Answer](#)

Answer: a

Explanation: Since, with each value of 'i' the value of 'j' is executed three times i.e for i = 0, j = 0, 0, 0, i = 1, j = 2, 2, 2.

Output:

0, 0, 0 4, 4, 4 8, 8, 8

**217. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
1. static void Main(string[] args)
2. {
3.     char A = 'K';
4.     char B = Convert.ToChar(76);
5.     A++;
6.     B++;
7.     Console.WriteLine(A+ " " +B);
8.     Console.ReadLine();
9. }
```

a) M L

b) U L

c) L M

d) A B

[View Answer](#)

Answer: c

Explanation: "++" increments the value of character by 1. A and B are given values K and 76, when we use increment operator their values increments by 1, A and B becomes L and M.

Output:

L, M

**218. Hoàn thành mã C # sau với "điều kiện foreach".**

```
1. int[][] a = new int[2][];
2. a[0] = new int[3]{3, 4, 2};
3. a[1] = new int[2]{8, 5};
4. foreach( int[] i in a)
5. {
6.     /* add for loop */
```

```
7. console.write( j+ " ");
8. console.writeline();
9. }
```

- a) foreach (int j = 1;(j(<)(a(0).GetUpperBound)); (j++));
- b) foreach (int j = 1;(j(<)(a.GetUpperBound(0))); (j++));
- c) foreach (int j in a.Length);
- d) foreach (int j in i);

[View Answer](#)

Answer: d

Explanation: None.

**219. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
1. static void Main(string[] args)
2. {
3.     double a = 345.09;
4.     byte c = (byte) a;
5.     Console.WriteLine(c);
6.     Console.ReadLine();
7. }
```

- a)98
- b)89
- c)88
- d)84

[View Answer](#)

Answer: b

Explanation: Type casting a larger variable into a smaller variable results in modules of larger variable by range of smaller variable. a is '345.09' which is larger than byte range ie -128 to 127.

Output :

89

**230. Phát biểu nào đúng về mã c # .NET sau đây?**

```
int[] a= {11, 3, 5, 9, 6};
```

- a) 'a' is a reference to the array created on stack
- b) 'a' is a reference to an object created on stack
- c) 'a' is a reference to an object of a class that compiler derives from 'System.Array' class
- d) None of the mentioned

[View Answer](#)

Answer: c

Explanation: A perfect way of defining single array in C# which is derived automatically from class 'System.Array'.

**231. Lợi thế của việc sử dụng mảng răng cưa 2D so với mảng chữ nhật 2D là gì?**

- a) Easy initialization of elements
- b) Allows unlimited elements as well as rows which had '0' or are empty in nature
- c) All of the mentioned
- d) None of the mentioned

[View Answer](#)

Answer: b

Explanation: In many applications where 2 dimensional arrays are used, not all rows need to have all the elements i.e they are sparse. Many rows have 0 elements. In such cases it is better to use 2D jagged arrays as they allow unequal number of elements in each row and also allow for empty rows.

**233. Câu nào đúng về mã C # sau đây?**

```
int[, ]a={{5, 4, 3},{9, 2, 6}};
```

a) 'a' represents 1-D array of 5 integers

b) a.GetUpperBound(0) gives 9

c) 'a' represents rectangular array of 2 columns and 3 arrays

d) a.GetUpperBound(0) gives 2

[View Answer](#)

Answer: c

Explanation: By definition.

### 234. Đầu ra của đoạn mã C # sau đây sẽ là gì?

```
1. static void Main(string[] args)
2. {
3.     Program p = new Program();
4.     p.display(2, 3, 8);
5.     int []a = { 2, 56, 78, 66 };
6.     Console.WriteLine("example of array");
7.     Console.WriteLine("elements added are");
8.     p.display(a);
9.     Console.ReadLine();
10. }
11. public void display(params int[] b)
12. {
13.     foreach (int i in b)
14.     {
15.         Console.WriteLine("ARRAY IS HAVING:{0}", i);
16.     }
17. }
```

a) Compile time error

b) Run time error

c) Code runs successfully but prints nothing

d) Code runs successfully and prints given on console

[View Answer](#)

Answer: d

Explanation: Object 'p' makes a call to invoke function display() and hence consecutively prints the output. Array 'a' is declared with elements again object 'p' makes a call to display() and hence, consecutively prints the output with given elements.

Output:

```
ARRAY IS HAVING:2
ARRAY IS HAVING:3
ARRAY IS HAVING:8
elements added are:
ARRAY IS HAVING:2
ARRAY IS HAVING:56
ARRAY IS HAVING:78
ARRAY IS HAVING:66
```

### 235. Cách xác định và khởi tạo mảng 3 số nguyên nào là đúng?

a) int[] a={78, 54}

b)

```
int[] a;
a = new int[3];
a[1] = 78;
a[2] = 9;
a[3] = 54;
```

c)

```
int[] a;  
a = new int{78, 9, 54};
```

d)

```
int[] a;  
a = new int[3]{78, 9, 54};
```

[View Answer](#)

Answer: d

Explanation: None.

**236. Chọn sự khác biệt có chọn lọc giữa mảng trong c # và mảng trong ngôn ngữ lập trình khác.**

a) Declaring array in C# the square bracket([]) comes after the type but not after identifier

b) It is necessary to declare size of an array with its type

c) No difference between declaration of array in c# as well as in other programming languages

d) All of the mentioned

[View Answer](#)

Answer: a

Explanation: i. When declaring an array in C#, the square brackets ([]) come after the type, not the identifier. Brackets after the identifier are not legal syntax in C#.

example :

```
int[] IntegerArray;
```

ii. The size of the array is not part of its type as it is in the C language. This allows to declare an array and assign any array of int objects to it, regardless of the array's length.

```
int[] IntegerArray;  
IntegerArray = new int[10];  
IntegerArray = new int[50];
```

### Basic Operation on Strings

**237. Which of the following string() method are used to compare two strings with each other?**

a) CopyTo()

**b) Copy()**

c) Compare()

d) CompareTo()

[View Answer](#)

Answer: b

Explanation: Creates a new string by copying one string to another.

**238. Choose the base class for string() method:**

a) System.Array

b) System.char

**c) System.String**

d) None of the mentioned

[View Answer](#)

Answer: c

Explanation: String is an alias for the predefined "System.string" class from which most of the string() methods are derived.

**239. What will be the output of the following C# code?**

```
1. static void Main(string[] args)  
2. {  
3.     string s1 = " Cshr ";  
4.     string s2 = s1.Insert(3, " a ");  
5.     string s3 = s2.Insert(5, " p ");
```

```

6.   for (int i = 0; i < s3.Length; i++)
7.       Console.WriteLine(s3[i]);
8.       Console.ReadLine();
9.   }

```

a) Csharp

b) CsharpP

**c) Csharp**

d) Cshrap

[View Answer](#)

Answer: c

Explanation: Insertion of character 'a' at position '3' using insert() which returns a new string with a substring inserted at a specified location.

Output:

Csharp

**240. Which of the following statement is correct about a string in C#.NET?**

a) The System.Array class is used to represent a string

**b) A string has a zero-based index**

c) A number cannot be represented in the form of a string

d) A string is mutable because it can be modified once it has been created

[View Answer](#)

Answer: b

Explanation: None.

**241. What will be the output of the following C# code?**

```

1.   static void Main(string[] args)
2.   {
3.       string s1 = "Hello";
4.       string s2 = "hello";
5.       if (s1 == s2)
6.           Console.WriteLine("Equal");
7.       else
8.           Console.WriteLine("Unequal");
9.       if (s1.Equals(s2))
10.           Console.WriteLine("Equal");
11.       else
12.           Console.WriteLine("Unequal");
13.       Console.ReadLine();
14.   }

```

15. a)

16. Equal

17. Unequal

18.

b)

Unequal

Equal

c)

Equal

Equal

**d)****Unequal****Unequal**[View Answer](#)

Answer: d

Explanation: In first comparison it is being checked either two strings are equal or not but in second comparison it is checked whether two references are equal or not.

Output:

Unequal

Unequal

**242. What will be the output of the following C# code?**

```

1.   static void Main(string[] args)
2.   {

```

```
3. string s1 = "Hello" + " I " + "Love" + " ComputerScience ";  
4. Console.WriteLine(s1);  
5. Console.ReadLine();  
6. }
```

a) HelloILoveComputerScience

**b) Hello I Love ComputerScience**

c) Compile time error

d) Hello

[View Answer](#)

Answer: b

Explanation: Here '+' defined operator works as concatenation for strings.

Output :

Hello I Love ComputerScience

**243. Correct way to find if contents of two strings are equal?**

a) if (s1 = s2)

b) if (s1 != s2)

**c) if (strcmp (s1 ,s2))**

d) if ( s1 is s2)

[View Answer](#)

Answer: c

Explanation: "==" operator used to compare length of two strings and strcmp() is the inbuilt method derived from string class.

**244. Which of the following statements are correct?**

a) String is value type

**b) String literals can contain any character literal including escape sequences**

c) The equality operators are defined to compare values of string objects as well as references

d) All of the mentioned

[View Answer](#)

Answer: b

Explanation: None

**245. Which of these operators can be used to concatenate two or more String objects?**

**a) +**

b) +=

c) &

d) ||

[View Answer](#)

Answer: a

Explanation:

```
string s1 = "Hello" + " I " + "Love" + " ComputerScience ";  
Console.WriteLine(s1);  
Hello I Love ComputerScience.
```

**246. The Method use to remove white space from a string?**

a) Split()

b) Substring()

**c) Trim()**

d) TrimStart()

[View Answer](#)

Answer: c

Explanation: Perfectly removes whitespace from string whereas TrimStart() removes a string of characters from the end of the string.



**String Class with Description****247. Chuỗi String trong C # nghĩa là gì?**

- a) Một biến số
- b) Mảng ký tự
- c) Đối tượng**
- d) Lớp

**248. Thuật ngữ ' immutable ' có nghĩa là gì đối với các đối tượng chuỗi?**

- a) Chúng tôi có thể sửa đổi các ký tự có trong chuỗi
- b) Chúng tôi không thể sửa đổi các ký tự có trong chuỗi**
- c) Chúng tôi không thể thực hiện nhiều thao tác so sánh, chèn, nối,..
- d) Không có điều nào được đề cập

**249. Để thực hiện thao tác so sánh trên các chuỗi được hỗ trợ, các thao tác được hỗ trợ là \_\_\_\_\_**

- a) So sánh ()
- b) Bằng ()
- c) Phép toán '==' gán
- d) Tất cả những điều đã đề cập**

**250. Đầu ra của đoạn mã C # sau đây sẽ là gì?****static void Main(string[] args)**

```
{  
    string s1= "Hello I Love Csharp";  
    Console.Write( Convert. Tochar( (s1.IndexOf('I')      s1.IndexOf('l'))*s1.IndexOf('p')));  
    Console.ReadLine();  
}
```

- a) I
- b) Hello I
- c) Love
- d) H**

**251. Cách đúng để chuyển một chuỗi thành chữ hoa bằng cách sử dụng phương thức lớp string ()?**

- a) Upper ()
- b) ToUpper ()
- c) Object.ToUpper ()**
- d) Không có điều nào được đề cập

**252. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
1. static void Main(string[] args)  
2. {  
3.     String obj = "hello";  
4.     String obj1 = "world";  
5.     String obj2 = obj;  
6.     Console.WriteLine (obj.Equals(obj2) + " " + obj2.CompareTo(obj) );  
7.     Console.ReadLine();  
8. }
```

- a) True True
- b) False False

**c) True 0**

d) False 1

**253. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
1. static void Main(string[] args)
2. {
3.     String obj = "hello";
4.     String obj1 = "world";
5.     String obj2 = obj;
6.     Console.WriteLine(obj + " " + obj1);
7.     string s = obj + " " + obj1;
8.     Console.WriteLine(s.Length);
9.     Console.ReadLine();
10. }
```

a. hello world  
10b. hello world  
6**c. hello world****11**d. hello world  
5**254. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
1. static void Main(string[] args)
2. {
3.     String obj = "hello";
4.     String obj1 = "world";
5.     String obj2 = obj;
6.     string s = obj+" "+obj1;
7.     Console.WriteLine(s.IndexOf('r'));
8.     Console.ReadLine();
9. }
```

a)7

**b)8**

c)9

d) 10

**255. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    String obj = "hello";
    String obj1 = "world";
    String obj2 = obj;
    string s = obj + " " + obj1;
    Console.WriteLine(s.Substring(6,5));
    Console.ReadLine();
}
```

a) hello

b) orld

**c) world**

d) o world

**256. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
1. static void Main(string[] args)
2. {
3.     String obj = "hello";
4.     String obj1 = "worn";
5.     String obj2 = obj;
6.     Console.WriteLine(obj + " " + (obj1.Replace('w', 'c')));
7.     Console.ReadLine();
8. }
```

- a) hello hello
- b) hello worn
- c) hello corn**
- d) hello

**Searching and Modifying Strings****257. Phương thức nào trong số các phương thức này của lớp String được sử dụng để tách một chuỗi con khỏi một đối tượng String?**

- a) substring()
- b) Substring()**
- c) SubString()
- d) Không có chuỗi nào được đề cập

Xem câu trả lời

Đáp án: b

Giải thích: Không có.

**258. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    String a = "Ilove";
    String b = "CSHARP";
    b = string.Concat(a, ' ', b);
    Console.WriteLine(b);
    Console.ReadLine();
}
```

- a) IloveCSHARP
- b) I loveCSHARP
- c) Ilove
- d) Ilove CSHARP**

Xem câu trả lời

Đáp án: d

Giải thích: Phương thức Concat () được sử dụng để nối hai chuỗi mà không cần sử dụng toán tử '+'.  
Đầu ra:

Ilove CSHARP

**259. Phương thức nào trong số các phương thức này của lớp được sử dụng để loại bỏ các khoảng trắng phía trước và phía sau?**

- a) startedWith ()
- b) trim ()
- c) Trim ()**
- d) doTrim ()

Xem câu trả lời

Đáp án: c

Giải thích: Không có.

**260. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    String a = "Ilove";
    String b = "CSHARP";
    b = string.Concat(a, ' ', b);
    string d = b.TrimStart('I', 'l', 'o', 'H');
    Console.WriteLine(d);
    Console.ReadLine();
}
```

a) Ilove CSHARP

b) love CSHARP

c) ve CSHARP

d) ve CSARP

Xem câu trả lời

Đáp án: c

Giải thích: trimstart () xóa ký tự được đề cập liên tiếp ở các vị trí phía trước chứ không phải các ký tự được đề cập ở giữa các vị trí.

Đầu ra:

ve CSHARP

**261. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    String c = " Hello Computer ";
    String a = c.Trim();
    Console.WriteLine("\"" + s + "\"");
}
```

a) "Hello Computer"

b) "HelloComputer"

c) "Hello Computer"

d) Hello Computer

Xem câu trả lời

Đáp án: c

Giải thích: Phương thức Trim () được sử dụng để loại bỏ khoảng trắng tiến và lùi trong chuỗi.

Đầu ra:

"Hello Computer"

**262. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    String a = "Ilove";
    String b = "CSHARP";
    b = string.Concat(a, ' ', b);
    string d = b.TrimStart('I', 'l', 'o', 'H');
    Console.WriteLine(d);
    Console.ReadLine();
}
```

a) "Hello Bye"

b) "HelloBye"

c) Hello Bye

d) HelloBye

Xem câu trả lời

Đáp án: d

Giải thích: Phương thức toán tử '+' hoạt động dưới dạng phương thức nối () và do đó được sử dụng để nối hai chuỗi lại với nhau.

Đầu ra:

Chào tạm biệt

**263. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    String c = "Hello";
    String a ;
    a = c.Replace('l', 'w');
    Console.WriteLine(a);
    Console.ReadLine();
}
```

a) Helloll

b) Hewlo

c) Helwo

d) Hewwo

Xem câu trả lời

Đáp án: d

Giải thích: Không có.

Đầu ra:

Hewwo

**264. Mệnh đề nào sau đây đúng?**

a) phương thức replace() thay thế lần xuất hiện cuối cùng của một ký tự khi gọi chuỗi bằng một ký tự khác

b) phương thức Replace () chỉ thay thế lần xuất hiện đầu tiên của một ký tự khi gọi chuỗi bằng một ký tự khác

c) Phương thức Replace () thay thế tất cả các lần xuất hiện của một ký tự trong việc gọi chuỗi với một ký tự khác

d) không có nào được đề cập

Xem câu trả lời

Đáp án: c

Giải thích: Theo định nghĩa.

**265. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    String c = "Hello i love you";
    String a ;
    a = c.Substring(12, 3);
    Console.WriteLine(a);
    Console.ReadLine();
}
```

a) ove

b) you

c) yo

d) love you

Xem câu trả lời

Đáp án: c

Giải thích: Không có.

Đầu ra:

Yo

**266. Cách nào sau đây là đúng để tìm ra chỉ số của 'giây' trong chuỗi "Cô ấy đã bán nhan sắc của mình trong một đêm cho người khác"?**

a)

```
String a = "She sold her beauty in one night to someone else";  
int i;  
i = a.SecondIndexOf("s");
```

b)

```
String a = "She sold her beauty in one night to someone else";  
int i, j;  
i = a.FirstIndexOf("s");  
j = a.IndexOf("s", i + 1);
```

c)

```
String a = "She sold her beauty in one night to someone else";  
int i, j;  
i = a.IndexOf("s");  
j = a.IndexOf("s", i + 1);
```

d) Không có điều nào được đề cập

Xem câu trả lời

Đáp án: c

Giải thích:

```
static void Main(string[] args)  
{  
    String c = "She sold her beauty in one night to someone else";  
    int i, j;  
  
    i = c.IndexOf("s");  
    j = c.IndexOf("s", i + 1);  
    Console.WriteLine(i + " " + j);  
    Console.ReadLine();  
}
```

Output : 4, 36

### Operation on Characters

**267. Phương thức nào trong số các phương thức này của lớp String được sử dụng để lấy độ dài của đối tượng String?**

a) get()

b) Sizeof()

c) lengthof()

d) length()

[View Answer](#)

Answer: d

Giải thích: Phương thức length () của lớp string được sử dụng để lấy độ dài của đối tượng dưới dạng chuỗi. Độ dài và do đó gọi phương thức length ().

**268. Phương thức nào sau đây là phương thức thay thế cho getChars () lưu trữ các ký tự trong một mảng byte?**

a) getBytes()

b) GetByte()

c) giveByte()

d) Give Bytes()

[View Answer](#)

Answer: a

Giải thích: `getBytes ()` lưu trữ ký tự trong một mảng byte. Nó sử dụng ký tự mặc định để chuyển đổi byte.

**269. Phương thức nào sau đây có thể được sử dụng để chuyển đổi tất cả các ký tự trong một chuỗi thành một mảng ký tự?**

- a) `CharAt()`
- b) `getChars()`
- c) `ToCharArray()`**
- d) All of the mentioned

[View Answer](#)

Answer: c

Giải thích: Không có.

**270. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
1. static void main(String args[])
2. {
3.     char chars[] = {'x', 'y', 'z'};
4.     String s = new String(chars);
5.     Console.WriteLine(s);
6. }
```

- a) x
- b) xy
- c) z
- d) xyz**

[View Answer](#)

Answer: d

Giải thích: Chuỗi (ký tự) là một phương thức khởi tạo của chuỗi lớp, nó khởi tạo chuỗi s với các giá trị được lưu trữ trong các ký tự mảng ký tự, do đó s chứa "xyz".

Output :

xyz

**271. Chọn phương thức `stringBuilder` hiệu quả giúp tạo đầu ra cho mã C # sau đây?**

```
1. static void Main(string[] args)
2. {
3.     StringBuilder s = new StringBuilder("object");
4.     s./ * _____ */ ("Oriented Language");
5.     Console.WriteLine(s);
6.     Console.ReadLine();
7. }
8. Output : objectOriented Language
```

- a) `Insert()`
- b) `Add()`
- c) `Append()`
- d) `Join()`

[View Answer](#)

Answer: c

Explanation:

```
static void Main(string[] args)
{
    StringBuilder s = new StringBuilder("object");
    s.Append("Oriented Language");
    Console.WriteLine(s);
    Console.ReadLine();
}
```

```
}  
Output : objectOriented Language
```

**272. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
1. static void Main(string[] args)  
2. {  
3.     string s = " i love you";  
4.     Console.WriteLine(s.IndexOf('l') + "    " + s.LastIndexOf('o') + "    " +  
       s.IndexOf('e'));  
5.     Console.ReadLine();  
6. }
```

- a) 3 5 7
- b) 4 5 6
- c) 3 9 6
- d) 2 4 6

[View Answer](#)

Answer: **c**

Giải thích: `indexOf('l')` và `lastIndexOf('o')` là các hàm được định nghĩa trước được sử dụng để lấy chỉ số của lần xuất hiện đầu tiên và cuối cùng của ký tự được trả bởi l và o tương ứng trong mảng đã cho.

Output :

**3, 9, 6**

**273. Các phương thức sau đây của Chuỗi lớp được sử dụng để trích xuất tất cả các ký tự từ một Chuỗi đối tượng**

- a) `CHARAT()`
- b) `Remove()`
- c) `charAt()`
- d) `Replace()`

[View Answer](#)

Answer: **b**

Giải thích: `Replace()` thay thế tất cả các trường hợp của một ký tự bằng một ký tự mới trong khi Xóa trích xuất các ký tự khỏi chuỗi.

**274. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
1. static void Main(string[] args)  
2. {  
3.     string c = "hello";  
4.     string c1 = c.Remove(1);  
5.     Console.WriteLine(c1);  
6.     Console.ReadLine();  
7. }
```

- a) ello
- b) h
- c) hell
- d) none of the mentioned

[View Answer](#)



Answer: **b**

Giải thích: Loại bỏ () xóa các ký tự khỏi chuỗi ngoại trừ ký tự được chỉ định với vị trí nhất định của nó.

Output :

h

**275. Một chuỗi thường được xử lý như thế nào?**

- a) On a character by character basis
- b) On a string by string basis
- c) Both On a character by character basis & On a string by string basis
- d) None of the mentioned

View Answer

Answer: **a**

Giải thích: Không có.

**276. Làm cách nào để in \\ trên màn hình?**

- a) Console.WriteLine("\\");
- b) Console.WriteLine("\\\\");
- c) Console.WriteLine("\\\\");
- d) Console.WriteLine("\\\\\\\\");

View Answer

Answer: **c**

Giải thích: Console.WriteLine ("\\\\");

Output :

\\

**Public & Private Access Modifier**

**277. Cái nào trong số này được sử dụng làm bộ chỉ định mặc định cho một thành viên của lớp nếu không có bộ xác định truy cập nào được sử dụng cho nó?**

- a) private
- b) public
- c) public, within its own class
- d) protected

Trả lời: a

a) riêng tư

**278. Từ khóa truy cập nào sau đây được sử dụng để truy cập các thành viên của lớp trước khi đối tượng của lớp đó được tạo?**

- a) public
- b) private
- c) static
- d) protected

Trả lời: c

c) tĩnh

**279. Lớp cơ sở nào có thể truy cập được đối với các thành viên lớp dẫn xuất?**

- a) static
- b) protected
- c) private
- d) Shared

Trả lời: b

b) được bảo vệ

**280. Quá trình mà chúng ta có thể kiểm soát các phần của một chương trình có thể truy cập vào các thành viên của một lớp là gì?**

a) Polymorphism

b) Abstraction

**c) Encapsulation**

d) Recursion

Trả lời: c

**c) Tính đóng gói**

**281. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
class sum
{
    public int x;
    private int y;
    public void math(int a, int b)
    {
        x = a * 4;
        y = b;
    }
}

class Program
{
    static void Main(string[] args)
    {
        sum p = new sum();
        p.math(12, 30);
        Console.WriteLine(p.x + " " + p.y);
        Console.ReadLine();
    }
}
```

a) 48, 30

b) 48, 0

c) 0, 0

**d) Compile time error**

Giải thích: biến 'y' không thể truy cập được do mức độ truy cập của nó.

Đầu ra:

Change **private** y to **public** y

**282. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
class sum
{
    public int x;
    public int y;
    public int add (int a, int b)
    {
        x = a + b;
        y = x + b;
        return 0;
    }
}

class Program
{
    static void Main(string[] args)
    {
```

```
sum obj1 = new sum();
sum obj2 = new sum();
int a = 2;
obj1.add(a, a + 1);
obj2.add(5, a);
Console.WriteLine(obj1.x + " " + obj2.y);
Console.ReadLine();
}
```

a) 6, 9

**b) 5, 9**

c) 9, 10

d) 3, 2

Xem câu trả lời

Đáp án: b

Giải thích: Ở đây,  $a = 2$ ,  $a + 1 = 2 + 1 = 3$ .Vậy,  $a = 2$ ,  $b = 3$ . $x = 2 + 3 = 5$ . $y = 5 + 3 = 8$ .Tương tự,  $a = 5$ ,  $b = a + 1 = 4$ . $y = 5 + 4 = 9$ .**283. Đầu ra của đoạn mã C # sau đây sẽ là gì?**

```
class math
{
    public int a,b;
    public math(int i, int j)
    {
        a = i;
        b = j;
    }
    public void sum(math m)
    {
        m.a *= 2;
        m.b += 2;
    }
}
class Program
{
    static void Main(string[] args)
    {
        math t = new math(20, 10);
        t.sum(t);
        Console.WriteLine(t.a + " " + t.b);
        Console.ReadLine();
    }
}
```

a) 10, 20

b) 20, 10

**c) 40, 12**

d) 5, 40

Đáp án: c

Giải thích: `t.sum(t)` gửi đối tượng 't' dưới dạng tham số mà các biến a & b được nhân và cộng với 2 tương ứng bởi hàm `sum()` của toán lớp. Do đó, a & b lần lượt trở thành 40 và 12.

40, 12

**284. Công cụ sửa đổi truy cập được định nghĩa trong một lớp là?**

- a) public, private, protected
- b) public, internal, protected internal
- c) public, private, internal, protected internal

**d) public, private, protected, internal, protected internal**

Trả lời: d

**285. Chọn câu nào sai về bản chất?****a) The base class member functions can access public member functions of derived class**

- b) An object of a derived class cannot access private member of the base class
- c) Private members of the base class cannot be accessed by derived class member functions or objects of derived class
- d) None of the mentioned

Trả lời: a

**a) Các hàm thành viên lớp cơ sở có thể truy cập các hàm thành viên công khai của lớp dẫn xuất****286. Bộ chỉ định truy cập nào trong số này phải được sử dụng cho phương thức main ()?****a) private**

- b) public
- c) protected
- d) none of the mentioned

Trả lời: a

Giải thích: Theo mặc định, main () được khai báo là private nếu không có trình chỉ định truy cập nào khác được sử dụng cho nó.

### Use of Ref and Out Parameters

**286. Đầu ra của mã C # sau đây sẽ là gì?**

```
class Program
{
    static void Main(string[] args)
    {
        int i = 5;
        int j;
        method1(ref i);
        method2(out j);
        Console.WriteLine(i + " " + j);
    }
    static void method1(ref int x)
    {
        x = x + x;
    }
    static void method2(out int x)
    {
        x = 6;
        x = x * x;
    }
}
```

a) 36, 10

**b) 10, 36**

- c) 0, 0  
d) 36, 0

**287. Các câu lệnh về từ khóa 'ref' được sử dụng trong C # .NET là?**

- a) The ref keyword causes arguments to be passed by reference  
b) While using 'ref' keyword any changes made to the parameter in the method will be reflected in the variable when control is passed back to the calling method  
c) Ref usage eliminates overhead of copying large data items

**d) All of the mentioned**

**288. Đầu ra của mã C # sau đây sẽ là gì?**

```
static void main(string[] args)
{
    int n = 1;
    method(n);
    console.WriteLine(n);
    method1(ref n);
    console.WriteLine(n);
}

static void method(int num)
{
    num += 20;
    console.WriteLine(num);
}

static void method1(ref int num)
{
    num += 20;
    console.WriteLine(num);
}
```

|    |    |    |    |
|----|----|----|----|
| a) | b) | c) | d) |
| 1  | 21 | 11 | 21 |
| 1  | 1  | 21 | 1  |
| 1  | 21 | 21 | 21 |
| 1  | 21 | 11 | 21 |

**289. Phương pháp nào sau đây giải thích mã C #?**

```
static void Main(string[] args)
{
    int a = 10, b = 20;
    method(ref a, ref b);
    console.WriteLine(a + " " + b);
}

static void swap(ref int i, ref int j)
{
    int t;
    t = i;
    i = j;
    j = t;
}
```

**a) Call by reference**

b) Call by value

- c) Output parameter
- d) parameter arrays

**290. Đầu ra của mã C # sau đây sẽ là gì?**

```
static void main(string[] args)
{
    int []arr = new int[]{ 1, 2, 3, 4, 5};
    fun (ref arr);
    for (int i = 0; i < arr.Length ; i++)
        Console.WriteLine( arr[i] + ", ");
}
static void fun(ref int[]a)
{
    a = new int[6];
    a[3] = 32;
    a[1] = 24;
}
```

a) 0, 0, 32, 0, 0, 0

**b) 0, 24, 0, 32, 0, 0**

c) 24, 0, 32, 0, 0, 0

d) 0, 0, 32, 0, 0, 0

**291. Đầu ra của mã C # sau đây sẽ là gì?**

```
static void main(string[] args)
{
    int i;
    int res = fun (out i);
    console.writeline(res);
    console.readline();
}
static int fun(out int i)
{
    int s = 1;
    i = 7;
    for (int j = 1; j <= i; j++ )
        s = s * j;
    return s;
}
```

a) 4490

**b) 5040**

c) 5400

d) 3500

**292. Đầu ra của mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    int a = 5;
    int b = 0, c = 0;
    method (a, ref b, ref c);
    Console.WriteLine(b + " " + c);
    Console.ReadLine();
}
```

```
static int method(int x, int p, ref int k)
{
    p = x + x * x;
    k = x * x + p;
    return 0;
}
```

- a) 30, 55
- b) 55, 30
- c) Compile time error
- d) 0, 0

**293. Từ khóa được sử dụng để xác định cuộc gọi theo tham số tham chiếu trong C # .NET?**

- a) &
- b) out
- c) ref
- d) &&

**294. Chọn sự phù hợp chính xác của khai báo tham số.**

```
static Void main(string[] args)
{
    int a = 5;
    int b = 6;
    float c = 7.2f;
    math (ref a, ref b, ref c);
    Console.WriteLine(a + " " + b + " " + c);
}
static int math(/*add parameter declaration */)
{
    a += b;
    b *= (int)c;
    c += a * b;
    return 0;
}
```

- a) ref int a, int b, ref float c
- b) ref int a, ref float c, ref int b
- c) ref int a, ref int b, float c
- d) ref int a, ref int b, ref float c

**295. Câu nào đúng / đúng?**

- a) An argument passed to a ref parameter need not be initialized first
- b) Variables passed as out arguments need to be initialized prior to being passed
- c) To use a ref parameter, only the calling method must explicitly use the ref keyword
- d) None of the mentioned

#### Use of Variable Number of Arguments

**296. Phương thức xử lý số lượng đối số lớn hoặc thay đổi được gọi là.....?**

- a) Value parameters
- b) Output parameters
- c) Parameter arrays
- d) None of the mentioned

**297. Các công cụ sửa đổi được sử dụng để xác định một mảng tham số hoặc danh sách các đối số là?**

- a) ref
- b) out
- c) param
- d) var

298. Đầu ra của mã C # sau đây sẽ là gì?

```
static void Main(string[] args)
{
    object[] a = { " 1 ", 4.0f, " harsh "};
    fun(a);
    Console.ReadLine();
}
static void fun(params object[] b)
{
    for (int i = 0; i < b.Length - 1; i++)
        Console.WriteLine(b[i] + " ");
}
```

- a) 1 4.0 harsh
- b) 1 4
- c) 1 4 hars
- d) 1 4 harsh

299. Mệnh đề nào sau đây đúng?

- a) C SHARP allows a function to have arguments with default values
- b) C SHARP allows a function to have variable number of arguments
- c) Params are used to specify the syntax for a function having arguments
- d) Omitting the return value type in method definition results into an exception

300. Đầu ra của mã C # sau đây sẽ là gì?

```
static void Main(string[] args)
{
    int [] a = {1, 2, 3, 4, 5};
    fun(a);
    Console.ReadLine();
}
static void fun(params int[] b )
{
    int[] k = { 3, 4, 7, 8, '\0' };
    for (int i = 0; i < b.Length; i++)
    {
        b[i] = b[i] + k[i] ;
        Console.WriteLine( b[i] + " ");
    }
}
```

- a) Compile time error
- b) 3, 4, 7, 8, 5
- c) 3, 4, 7, 8, 5, 1, 2, 3, 4, 5
- d) 4, 6, 10, 12, 5

301. Đầu ra của mã C # sau đây sẽ là gì?

```
static void Main(string[] args)
{
    int [] a = {1, 2, 3, 4, 5};
    fun(a);
    Console.ReadLine();
}
```



```

}
static void fun(params int[] b)
{
    for (int i = 0; i < b.Length; i++)
    {
        b[i] = b[i] * 5;
        Console.WriteLine(b[i] + "");
    }
}

```

1, 2, 3, 4, 5

b) 5, 10, 15, 20, 25

c) 5, 25, 125, 625, 3125

d) 6, 12, 18, 24, 30.

**302. Đầu ra của mã C # sau đây sẽ là gì?**

```

static void Main(string[] args)
{
    int[] a = { 2, 21, 34, 46, 85, 88, 90 };
    fun(a);
    Console.WriteLine(a + " ");
    Console.ReadLine();
}
static void fun(params int [] b)
{
    int [] c = { 1, 2, 3, 4, 5, 6, 7 };
    int i;
    for (i = 0; i < b.Length; i++)
        if (b[i] % 2 == 0)
        {
            c[i] = b[i];
        }
    Console.WriteLine("even numbers are:");
    for (i = 0; i <= b.Length; i++)
    {
        Console.WriteLine(c[i]);
    }
}

```

a) Compile time error

b) 2, 21, 34, 4, 6, 46, 88, 90

c) 2, 4, 34, 46, 6, 88, 90

d) 2, 34, 46, 88, 90

**303. Chọn khai báo đúng của mảng tham số xác định.**

```

a)
void func(int[] x)
{
    }

b)
void func(int x)
{
    }

```

c)  
 void func(param int[])  
 {  
  
 }  
 }

d)  
 void fun(param int[] x)  
 {  
  
 }

**304. Đầu ra của mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    int[] x = { 80, 82, 65, 72, 83, 67 };
    fun(x);
    Console.ReadLine();
}

static void fun(params int [] b )
{
    int i;
    for (i = 5; i >= 0 ; i--)
    {
        Console.WriteLine(Convert.ToChar(b[i]));
    }
}
```

a) 67 83 72 65 82 80

b) P R A H S C

c) C S H A R P

d) 80 82 65 72 83 67

**305. Đầu ra của mã C # sau đây sẽ là gì?**

```
static void Main(string[] args)
{
    int[] x = {65, 66, 67, 68, 69, 70};
    fun(x);
    Console.ReadLine();
}

static void fun(params int[] b )
{
    int i;
    for (i = 5; i > 0 ; i--)
    {
        b[i] = b[i] + 32;
        Console.WriteLine(Convert.ToChar(b[i]));
    }
}
```

a) A, B, C, D, E, F

b) F, E, D, C, B, A

c) f, e, d, c, b

d) b, c, d, e, f