

## 프로그래밍언어론

문 1. 다음 웹문서가 성공적으로 웹 브라우저에 나타났다고 할 때 웹 브라우저에 나타나는 내용은?

```
<html><head><title>ScriptTest</title></head>
<body><script type="text/javascript" language="JavaScript">
    <!--
    var count = 0;
    while (count < 10) {
        document.write(count);
        count++;
    }
    //--!>
</script></body></html>
```

- ① 0123456789                      ② 12345678910  
③ 024681012141618              ④ \*\*\*\*\*

문 2. 아래 HTML 문서의 웹 브라우저 출력 결과로 옳은 것은?

```
<html><head><title>Items</title></head><body>
    <ol>
    <li>First</li>
    <li>Second</li>
    <ul>
    <li>Third</li>
    <li>Fourth</li>
    </ul>
    <li>Fifth</li>
    </ol>
</body></html>
```

- ① ○ First                      ② 1. First  
    ○ Second                  2. Second  
    ○ Third                    3. Third  
    ○ Fourth                  4. Fourth  
    ○ Fifth                    5. Fifth  
③ 1. First                      ④ 1. First  
    2. Second                  2. Second  
        ○ Third                  ○ Third  
        ○ Fourth                  ○ Fourth  
    3. Fifth                    5. Fifth

문 3. C++에서 원소 자료형이 double인 STL 벡터(vector)를 바르게 선언한 문장은? (단, #include<vector>와 using namespace std;는 미리 선언되어 있다고 가정하자)

- ① vector(double) v;              ② vector<double> v;  
③ vector double v;              ④ vector v[double];

문 4. C++에서 매개변수의 개수와 타입을 바꾸어 같은 이름의 함수를 선언하는 기능은?

- ① 오버라이딩(overriding)          ② 시스템콜(system call)  
③ 오버로딩(overloading)          ④ 가상함수(virtual function)

문 5. 다음 자바 프로그램의 실행결과로 옳은 것은?

```
class ParaPassing {
    public void change(int i, int[] j) {
        i = 20; j[3] = 400;
    }
    public void display(int i, int[] j) {
        System.out.println("i: "+i);
        System.out.print("j: ");
        for (int k = 0; k < j.length; k++)
            System.out.print(j[k]+" ");
        System.out.println();
    }
}

public class ParaPassingTest {
    public static void main(String[] args) {
        ParaPassing pp = new ParaPassing();
        int i = 10, j[] = { 1, 2, 3, 4 };
        pp.change(i, j);
        pp.display(i, j);
    }
}
```

- ① i: 10                              ② i: 20  
    j: 1 2 3 400                      j: 1 2 3 400  
③ i: 10                              ④ i: 20  
    j: 1 2 3 4                          j: 1 2 3 4

문 6. C++에서 메소드(method)는 virtual로 선언된 경우에만 동적 바인딩을 허용한다. 그렇다면 다음 중 Java에서 메소드의 동적 바인딩을 허용하는 경우는?

- ① final로 정의된 메소드  
② static으로 정의된 메소드  
③ private으로 정의된 메소드  
④ final, static, private으로 정의되지 않은 메소드

문 7. 다음 Java 프로그램의 실행 결과로 옳은 것은?

```
import java.util.*;
public class List {
    public static void main(String[] args) {
        LinkedList<Integer> ds = new LinkedList<Integer>();
        ds.addLast(new Integer(10));
        ds.addLast(new Integer(30));
        ds.addLast(new Integer(20));
        while (! ds.isEmpty())
            System.out.print(ds.removeLast()+" ");
        System.out.println();
    }
}
```

- ① 10 20 30                          ② 20 30 10  
③ 30 20 10                          ④ 10 30 20

문 8. Visual Basic 6.0의 프로시저에 대한 설명으로 옳은 것은?

- ① 호출한 프로시저로 결과 값을 반환하는 프로시저를 Sub 프로시저라 하고 결과 값을 반환하지 않는 프로시저를 Function 프로시저라 한다.
- ② 프로시저의 반환값이 있다고 하더라도 'As 자료형'을 생략할 수 있다.
- ③ 매개변수를 ByRef로 선언하면 변경된 매개변수의 값을 호출한 프로시저로 전달받을 수 없다.
- ④ 매개변수 전달 방식을 명시하지 않은 경우에 기본적으로 ByVal 방식으로 전달된다.

문 9. 다음은 Visual Basic 6.0 프로그램이다. Command1 버튼을 세 번 클릭했을 때 나타나는 최종 출력결과는?

```
Option Explicit
Public a As Integer
Private b As Integer
Dim c As Integer
Private Sub Command1_Click()
    Dim d As Integer
    Static e As Integer
    a = a + 1
    b = b + 1
    c = c + 1
    d = d + 1
    e = e + 1
    Form1.Show
    Form1.Print a, b, c, d, e
End Sub
Private Sub Form_Load()
    a = 2
    b = 3
    c = 4
End Sub
```

- ① 5 6 7 1 1                      ② 5 6 7 1 3
- ③ 5 6 7 3 3                      ④ 3 3 3 3 3

문 10. 다음 C++ 프로그램은 가상함수(virtual function)를 사용하는 프로그램이다. 이 프로그램이 정상적으로 수행되었을 때의 실행 결과는?

```
#include <iostream>
using namespace std;
class A {
protected:
    int n;
public:
    A(): n(0) {}
    int getn() { ++n; return n; }
    virtual void print(int i) { cout << "A: " << i << endl; }
};
class B: public A {
public:
    int getn() { return n; }
    void print(int i) { cout << "B: " << i << endl; }
};
int main() {
    B b; A *p = &b;
    p->print(p->getn());
    return 0;
}
```

- ① A: 0                              ② A: 1
- ③ B: 0                              ④ B: 1

문 11. 다음은 Visual Basic 6.0 프로그램의 일부이다. 명령버튼 cmdButton이 클릭될 때 picBox에 출력되는 결과로 옳은 것은?

```
Private Sub cmdButton_Click()
    Dim strFirst As String, strMiddle As String, strLast As String
    strFirst = "Augusta"
    strMiddle = "Ada"
    strLast = "Byron"
    Call Initials(strFirst, strMiddle, strLast)
End Sub
Private Sub Initials(c As String, b As String, a As String)
    Dim strTheInitials As String
    strTheInitials = Left(a, 1) & Left(b, 1) & Left(c, 1)
    picBox.Print strTheInitials
End Sub
```

- ① AAB                              ② ABA
- ③ BAA                              ④ ABa

문 12. 크기가 큰 배열을 매개변수로 전달할 때, 메모리의 양이 가장 적게 사용되는 매개변수 전달방법은?

- ① 값 전달(call-by-value)
- ② 값-결과 전달(call-by-value-result)
- ③ 참조 전달(call-by-reference)
- ④ 결과 전달(call-by-result)

문 13. 컴파일러(compiler)와 인터프리터(interpreter)에 대한 설명으로 옳은 것은?

- ① 컴파일러는 입력 프로그램 전체를 번역하여 목적 프로그램을 만든 뒤 실행하고, 인터프리터는 입력 프로그램의 문장을 바로 해석하여 실행한다.
- ② 수행 속도는 컴파일러가 생성한 코드가 대체적으로 빠르지만 반복 부분이 많은 프로그램의 경우에는 인터프리터가 더 빠르다.
- ③ 요구되는 메모리 측면에서 볼 때 컴파일러가 인터프리터보다 더 적은 메모리를 요구한다.
- ④ 프로그래밍 실습 과정처럼 오류를 많이 포함한 짧은 프로그램을 빈번히 실행해야 하는 환경에서는 컴파일러가 인터프리터보다 더 적합하다.

문 14. 다음은 Visual Basic 6.0 프로그램의 일부이다. 동적배열을 사용하기 위해 빈칸 ㉠에 가장 적합한 키워드는?

```
Private Sub Command1_Click()
    Dim strarray() As String
    ( ㉠ ) strarray(5, 2)
    ( ㉠ ) strarray(10, 1)
End Sub
```

- ① SetData                          ② Array
- ③ Const                            ④ ReDim

문 15. 다음은 Visual Basic 6.0 프로그램의 일부이다.

```
Private Sub Form_Load()
    Dim i As Integer, j As Integer
    Dim S As Long, T As Long
    S = 0
    ( ㉠ )
    Label1.Caption = "Result = " & S
End Sub
```

Label1 캡션에 아래 연산의 수행 결과를 출력하기 위한 빈칸 ㉠에 적합한 코드는?

1+(1+2)+(1+2+3)+...+(1+2+3+...+10)

- ①
- ```
For i = 1 To 10
    For j = 1 To i
        If i Mod 2 = 1 Then
            S = S + j
        Else
            S = S - j
        End If
    Next j
Next i
```
- ②
- ```
For i = 1 To 10
    For j = 1 To i
        S = S + j
    Next j
Next i
```
- ③
- ```
For i = 1 To 10
    T = 1
    For j = 1 To i
        T = T * j
    Next j
    S = S + T
Next i
```
- ④
- ```
For i = 1 To 10
    For j = 1 To i
        S = S + i
    Next j
Next i
```

문 16. 프로그래밍 언어의 추상화에는 프로세스 추상화(process abstraction)와 데이터 추상화(data abstraction)가 있다. 프로세스 추상화와 가장 가까운 언어구조는?

- ① 부프로그램(subprogram)      ② 클래스(class)  
③ 객체(object)                      ④ 캡슐화(encapsulation)

문 17. 다음 ANSI C 프로그램에서 A 지점까지 실행하였을 때 발생하는 쓰레기(garbage)는?

```
#include <stdlib.h>
int main() {
    int *pi;
    {
        int i, j, *pj;
        pi = &i;
        pj = (int *) malloc(sizeof(int));
    }
    /* A 지점 */
    return 0;
}
```

- ① pi                                      ② \*pi  
③ pj                                      ④ \*pj

문 18. 다음과 같이 왼쪽 상단에는 A.html 내용, 왼쪽 하단에는 B.html 내용, 오른쪽에는 C.html 내용이 나오도록 프레임을 구성하고자 할 때, 가장 바르게 작성한 HTML 문서는?

A.html 내용	C.html 내용
B.html 내용	

- ①
- ```
<html>
<head><title>Test</title></head>
<frameset cols="50%, *">
    <frameset rows="50%, 50%">
        <frame src="A.html">
        <frame src="B.html">
    </frameset>
    <frame src="C.html">
</frameset>
</html>
```
- ②
- ```
<html>
<head><title>Test</title></head>
<frameset rows="50%, *">
    <frameset cols="50%, 50%">
        <frame src="A.html">
        <frame src="B.html">
    </frameset>
    <frame src="C.html">
</frameset>
</html>
```
- ③
- ```
<html>
<head><title>Test</title></head>
<frameset cols="50%, *">
    <frame src="A.html">
    <frameset rows="50%, 50%">
        <frame src="B.html">
        <frame src="C.html">
    </frameset>
</frameset>
</html>
```
- ④
- ```
<html>
<head><title>Test</title></head>
<frameset rows="50%, *">
    <frame src="A.html">
    <frameset cols="50%, 50%">
        <frame src="B.html">
        <frame src="C.html">
    </frameset>
</frameset>
</html>
```

문 19. C++ 언어의 레퍼런스에 대한 설명으로 옳지 않은 것은?

- ① 레퍼런스는 다른 변수의 별명이다.  
② 레퍼런스는 선언 시에 반드시 초기화하여야 한다.  
③ 레퍼런스를 사용하는 과정에서 참조하는 대상을 변경할 수도 있다.  
④ C++의 다른 자료형과 달리 레퍼런스는 배열 형태로 선언할 수 없다.

문 20. XML 표준 명세는 EBNF(Extended Backus-Naur Form)의 형식으로 정의할 수 있다. 다음은 EBNF로 작성된 XML 표준 명세를 일부 발췌하여 변형한 것이다. 이 EBNF 명세에 의해서 만들어질 수 없는 것은?

```
<document> ::= <prolog> <element>
<prolog> ::= [ <XMLDecl> ] [ <doctypeDecl> ]
<XMLDecl> ::= '<?xml' <VersionInfo> '?'
<VersionInfo> ::= 'version' '=' ' "' <VersionNum> ' "'
<VersionNum> ::= '1.' ( '0' | '1' ) { '1' | '2' | '3' }
<doctypeDecl> ::= '<!DOCTYPE' <Name> '>'
<element> ::= <EmptyElemTag> | <STag> <content> <ETag>
<STag> ::= '<' <Name> { <Attribute> } '>'
<Attribute> ::= <Name> '=' <AttValue>
<ETag> ::= '</' <Name> '>'
<EmptyElemTag> ::= '<' <Name> { <Attribute> } '/>'
<Name> ::= 'n'
<content> ::= 'cont'
<AttValue> ::= 'av'
```

- ① <?xml version = "1.1" ?> <n n=av />  
② <!DOCTYPE n> <n/>  
③ <?xml version = "1.123" ?> <n n=av n=av> cont </n>  
④ <!DOCTYPE n> <n> </n>