

프로그래밍언어론

문 1. 연산자 -와 *로 구성된 정수 계산식에서 -가 *보다 우선순위가 높고 -는 우결합성을 가진다고 가정하자. 다음 중 결과 값이 가장 큰 것은?

- ① 1-3-2*2
- ② 4-1*1-2
- ③ 7-5*2
- ④ 2*4-5

문 2. 다음 Java 프로그램의 실행 결과는?

```
interface B {
    int b=1;
    int f();
}

abstract class A implements B {
    public abstract void g(int i);
    public int f() { return b*2; }
}

class C extends A {
    public void g(int i) { System.out.println(i*2); }
    public int f() { return super.f()*2; }
}

class D extends C {
    public void g(int i) { System.out.println(i*3); }
}

public class Test {
    public static void main(String[] args) {
        C c = new D();
        c.g(c.f());
    }
}
```

- ① 1
- ② 2
- ③ 8
- ④ 12

문 3. 다음 프로그램은 재귀호출을 사용하여 작성한 C 프로그램이다. 실행 결과는?

```
#include <stdio.h>
void fun(int a, int b, int c)
{
    if(c!=0) {
        fun(b, a+b, c-1);
        printf("%d %d %d\n",a,b,c);
    }
}
int main(int argc, char* argv[])
{
    int i=1, j=1, k=3;
    fun(i,j,k);
    return 0;
}
```

- ① 3 5 0
2 3 1
1 2 2
- ② 2 3 1
1 2 2
1 1 3
- ③ 2 3 0
1 2 1
1 1 2
- ④ 1 1 3
1 2 2
2 3 1

문 4. 다음은 C++에서의 다형성에 관한 예제이다. 이 프로그램에 대한 설명으로 옳은 것은?

```
#include<iostream.h>
class Polygon {
    public: virtual void edges() { cout<< "NA "; } // ㉠
};
class Rectangle : public Polygon {
    public: void edges() { cout<< "4 "; }
};
class Triangle : public Polygon {
    public: void edges() { cout<< "3 "; }
};
class Varangle : public Polygon {
    public: void edges(int e) { cout<< "# "; }
};
int main(int argc, char* argv[])
{
    Polygon* p[3];
    p[0] = new Rectangle();
    p[1] = new Triangle();
    p[2] = new Varangle();
    for (int i = 0; i < 3; i++)
        p[i]->edges();
    return 0;
}
```

- ① 프로그램이 실행되면 "NA NA NA"가 출력된다.
- ② 프로그램이 실행되면 "4 3 #"가 출력된다.
- ③ ㉠에서 virtual을 제거하고 실행하면 "4 3 NA"가 출력된다.
- ④ ㉠에서 virtual을 제거하고 실행하면 "NA NA NA"가 출력된다.

문 5. 다음 C 프로그램의 실행 결과는?

```
#include <stdio.h>
int y = 3;
int sample()
{
    ++y;
    return(5);
}
int main(int argc, char* argv[])
{
    int x;
    x = sample() * 2;
    x = x + y;
    printf("%d \n", x);
    return 0;
}
```

- ① 13
- ② 14
- ③ 15
- ④ 오류메시지가 출력된다.

문 6. 통칭(generics) 혹은 템플릿(templates) 기능을 지원하지 않는 언어는? (단, 각 언어는 최근 버전을 기준으로 한다)

- ① Pascal
- ② C++
- ③ Java
- ④ Ada

문 7. 다음 Java 프로그램의 실행 결과는?

```
public class ex1 {
    public static void main(String args[]) {
        int i, a = 0;
        int b[] = new int[3];
        for(i=0; i<=2; i++) {
            System.out.println("i=" + i);
            try {
                if (i == 0) a = 10/0;
                if (i == 1 ) a = 1;
                if (i == 2) a = b[i+3];
            }
            catch (ArithmeticException e) {
                System.out.println("error - divide by zero");
            }
            catch (ArrayIndexOutOfBoundsException e) {
                System.out.println("error - out of bound");
            }
            finally {
                i++;
            }
        }
        System.out.println("a=" + a);
    }
}
```

- ① i=0
error - divide by zero
i=2
error - out of bound
a=1
- ② i=0
a=1
- ③ i=0
error - divide by zero
i=2
error - out of bound
a=0
- ④ i=0
error - divide by zero
i=1
i=2
error - out of bound
a=0

문 8. 다음은 배열의 주소를 매개변수로 사용하여 함수를 호출하는 C 프로그램이다. 프로그램의 실행 결과는?

```
#include <stdio.h>
void printtest(int *p, int size);
int main(int argc, char* argv[])
{
    int a[5] = {10, 20, 30, 40, 50};
    int size = sizeof(a)/sizeof(a[0]);
    printtest(a, size/2);
    return 0;
}
void printtest(int *p, int size)
{
    printf("%d \n", p[size]);
}
```

- ① 20
- ② 30
- ③ 40
- ④ 50

문 9. 32bit 컴퓨터에서 배열의 시작 주소를 7094이라 가정할 경우 다음 코드의 실행 결과는?

```
#include <stdio.h>
int main(int argc, char* argv[])
{
    int *a;
    int b[5] = {2, 4, 7};
    a = &b[0];
    printf("%d, %d, %d \n", *a+3, *(a+3), a+3);
    return 0;
}
```

- ① 5, 0, 7106
- ② 5, 7, 7100
- ③ 0, 7, 7106
- ④ 5, 0, 7100

문 10. 객체지향 언어 C++, Java 에 대한 설명으로 옳지 않은 것은?

- ① 함수 오버로딩을 위한 조건은 함수 이름은 동일하되, 전달인자 자료형이나 개수가 달라야 한다.
- ② 캡슐화는 논리적으로 관련된 상수, 타입, 변수 메소드 등을 하나의 새로운 개체로 집약시키는 기법이다.
- ③ 객체지향 언어 C++, Java에서 상속이 지원하는 장점은 재사용성이다.
- ④ 상속된 파생 클래스에서 기저 클래스에 없는 메소드를 새로 정의할 수 없다.

문 11. 다음의 C 프로그램의 실행 결과는?

```
#include <stdio.h>
int main(int argc, char* argv[])
{
    char string[2][10] = {"variable", "constant"};
    printf("%s, %c \n", string, *(string[0]+1));
    return 0;
}
```

- ① variable, a
- ② variable, v
- ③ constant, c
- ④ variable, o

문 12. 다음 Java 프로그램에 대한 설명으로 옳은 것은?

```
class IfElse {
    public static void main(String args[]) {
        long charge = 5000 ;
        String country = args[0];
        String city = args[1];
        if (country.equals("한국"))
            if(city.equals("제주도"))
                charge = 10000;
            else
                charge = 20000;
        System.out.println(charge);
    }
}
```

- ① country 값이 "한국" 이 아니고 해외국가 이름이면 "20000"이 출력된다.
- ② country 값이 "한국" 이고 city 값이 "제주도"가 아닌 다른 도시이면 charge 값은 "5000"이 된다.
- ③ "java IfElse 중국 상해"를 실행하면 "20000"이 출력 된다.
- ④ "java IfElse 한국 부산"을 실행하면 "20000"이 출력 된다.

문 13. C 프로그램의 구성에 대한 설명으로 옳지 않은 것은?

- ① 전처리기 지시자로 #define, #include 등을 사용한다.
- ② 함수의 시작과 끝을 알려주기 위해 기호 { 와 }가 사용된다.
- ③ 프로그램의 실행은 main()함수로부터 실행되며, main()함수의 ()안에는 인자가 올 수 없다.
- ④ C 언어에서 제공되는 기본 자료형은 변수의 이름으로 사용될 수 없다.

문 14. 다음 C++ 프로그램의 실행 결과는?

```
#include <iostream.h>
void ErrorHandler(int test)
{
    try {
        if (test) throw test;
        else throw "예외 발생";
    }
    catch(int i) {
        cout << "예외에서 잡은 숫자 : " << i << "\n";
    }
    catch(char *str) {
        cout << "예외에서 잡은 문자열 : " << str << "\n";
        throw;
    }
}

int main(int argc, char* argv[])
{
    try {
        ErrorHandler(1);
        ErrorHandler(0);
        ErrorHandler(2);
    }
    catch(char *) {
        cout << "메인함수에서 잡은 예외 \n";
    }
    return 0;
}
```

- ① 예외에서 잡은 숫자 : 1
예외에서 잡은 숫자 : 0
예외에서 잡은 숫자 : 2
- ② 예외에서 잡은 숫자 : 1
예외에서 잡은 문자열 : 예외 발생
예외에서 잡은 숫자 : 2
- ③ 예외에서 잡은 숫자 : 1
예외에서 잡은 문자열 : 예외 발생
메인함수에서 잡은 예외
- ④ 예외에서 잡은 숫자 : 1
예외에서 잡은 문자열 : 예외 발생
메인함수에서 잡은 예외
예외에서 잡은 숫자 : 2

문 15. 다음의 BNF(Backus-Naur Form) 문법에 의하여 생성되는 언어를 나타내는 정규표현(regular expression)으로 옳은 것은?

$\langle E \rangle \rightarrow a\langle E \rangle \mid b\langle E \rangle \mid a$

- ① (ab)*a
- ② (ab)*a
- ③ a*b*a
- ④ a*b*a

문 16. 다음 Java 프로그램의 실행 결과는?

```
class Super {
    int index = 1;
    public void printVal() {
        System.out.println("super");
    }
}

class Sub extends Super {
    int index = 2;
    public void printVal() {
        System.out.println("sub");
    }
}

public class Test2 {
    public static void main(String args[]) {
        Super sup1 = new Sub();
        System.out.println(sup1.index);
        sup1.printVal();
    }
}
```

- ① 1
super
- ② 1
sub
- ③ 2
super
- ④ 2
sub

문 17. 다음 C 프로그램의 실행 결과는?

```
#include <stdio.h>
int main(int argc, char* argv[])
{
    int a= 1, b, c, d;
    d = (b= a++, c = b+2);
    printf("a=%d, b=%d, c=%d, d=%d\n", a, b, c, d);
    return 0;
}
```

- ① a=2, b=1, c=3, d=3
- ② a=2, b=1, c=4, d=4
- ③ a=2, b=2, c=3, d=3
- ④ a=2, b=2, c=4, d=4

문 18. 매개변수 전달기법 중 참조에 의한 전달방법(Call By Reference)에 대한 설명으로 옳은 것은?

- ① 실제 값으로 피호출 함수의 형식 매개 변수에 전달하는 기법이다.
- ② 실 매개변수는 피호출 함수의 결과를 받기 위해서만 사용된다.
- ③ 호출 함수는 피호출 함수에 실 매개변수의 주소를 전달하는 기법이다.
- ④ 형식 매개변수가 사용될 때마다 매번 실 매개 변수를 다시 계산한다.

문 19. 다음 Java 프로그램의 실행 결과는?

```
public class ClassA {
    int tmp = 100;
    ClassA() {
        System.out.println("tmp = " + tmp);
        tmp = 200;
    }
    public static void main(String[] args) {
        ClassA inst = new ClassA();
        System.out.println("tmp = " + inst.tmp);
    }
}
```

- ① tmp = 100
- ② tmp = 200
- ③ tmp = 100
tmp = 200
- ④ tmp = 200
tmp = 100

문 20. 다음 C 프로그램의 실행 결과는?

```
#include <stdio.h>
int main(int argc, char* argv[])
{
    int x = 3; int y = 4; int z = 5 ;
    x = y;
    {
        int x = 7, y = 2;
        x = z;
        printf("x1=%d, y1=%d, z1=%d \n", x, y, z);
    }
    printf("x2=%d, y2=%d, z2=%d \n", x, y, z);
    return 0;
}
```

- ① x1=5, y1=2, z1=5
x2=4, y2=4, z2=5
- ② x1=5, y1=2, z1=4
x2=5, y2=4, z2=5
- ③ x1=4, y1=5, z1=5
x2=4, y2=5, z2=5
- ④ x1=5, y1=5, z1=5
x2=4, y2=4, z2=4