

1. parameter

a. pass by value b. pass by reference c. pass by name d. pass by result e. pass by value-result

- side effect 발생하는 것
- aliasing 발생하는 것
- java에서 사용하는 것
- IN / OUT / INOUT으로 구분

2. exception handling

3. dynamic chain을 사용하는 이유

4. static chain을 사용하는 이유

5. thread 생성하는 방법 - 빈칸 채우세요

```
Class MyThread [   ] Thread{  
    [           ]//필수로 구현해야 하는 것  
}  
...  
Thread myth = new MyThread();  
myth.[   ];
```

```
Class MyThread [   ] Runnable{  
    [           ]//필수로 구현해야 하는 것  
}  
...  
[   ] th = new MyThread();  
[   ] myth = new [   ](th);  
myth.[   ];
```

6. CPU는 []와 []로 구성되어있는데, each instruction is executed in CPU consisting of [] and [] cycle.

7. java에서 예외 관련 키워드 5가지 말하고 설명하세요..

8. java 기준, 예외 발생시키는 방법 두가지

9. Activity Record 가 static보다 stack-dynamic 일 때의 장점 2가지

10. 예외 발생시킬 때, 메시지를 전달하려면

```
Class MyExcetption extends Exception{  
    public MyException(String message){  
        //들어가야할 한문장  
    }  
}
```

11. Overloaded Subprograms을 구현하는 것 (2가지)

12. Collection Framework

- List 특징
- Set 특징

- Map 특징

13. Loop 종류 3가지 및 사용 예시

14. Thread 설명 중 옳은 것?

- a. It is a scheduling unit
- b. Lightweight Process
- c. 프로세스 간 커뮤니케이션에 사용된다.
- d. run 메소드는 스레드의 생성자이다.
- e. t1.join()은 스레드 종료에 사용된다.
- f. 여러 생성자를 가질 수 있다.
- g. 하나의 클래스에서 여러 스레드를 가질 수 있다.

15. cooperation, competition, race condition에 대해 설명하세요

16. Activation Record Instance에 포함되는 값을 순서대로 나열하고 설명하세요.

17. Exception Propagation이란?

18. checked Exception과 unchecked Exception에 대해 설명하고 사용자 정의 unchecked 예외를 만드는 방법?

19. Event Handling과 Exception Handling의 차이점 한가지

20. Event Handling Code

```
class FrameTest3{  
    public static void main(String args[]){  
        Frame f = new Frame("Login");  
        f.setSize(300,200);  
  
        //window에서 일어난 event를 handler에게 전달하기 위해 listener 구현하세요  
  
        f.setVisible(true);  
    }  
}
```

21. Class에 관련된 연산 4가지

22. function과 procedure의 차이점

23. JAVA GUI component에서 발생하는 걸 뭐라고 하나?

24. positional parameter와 keyword parameter의 정의 및 keyword parameter의 장단점