•注意事项：

1、（如果题目中要求配置 SSH，如果够时间），那么要在每台客户端中（无论 Linux 还是 Windows），都在（Linux：ssh_config）（Windows：c:\Users\[user]\.ssh\conf）配置文件中配置添加 Cisco 的算法密钥，让它们能够正常连接。

```
Kexalgorithms +diffie-hellman-group14-sha1
HostKeyAlgorithms +ssh-rsa
```

2、因为 CML2.9.1 新增了一些客户端，因为图标都一样，如果改了名，就看不出设备属于什么型号，因此在检查环境时，要点开查看每个设备、客户端是什么版本

3、配置 VPN 时，Tunnel 接口的 mtu 建议设置为 1400

4、如果 ospf 运行在 DMVPN tunnel 接口中，那么建议网络类型设置为 broadcast 广播（point-to-muiltpoint 也可以）；

如果 ospf 运行在 flex VPN（IKEv2 隧道）tunnel 中，并且子网是/30，那么建议网络类型设置为 point-to-point（点对点）。

如果 ospf 邻居建立在 vlan 中，那么网络类型建议设置为 broadcast（广播）如果 ospf 邻居建立在 vlan 中，并且该网段掩码为 /30，那么网络类型建议设置为 point-to-point（点对点）

5、spanning-tree root guard 要做在 DS 交换机 连接 其他非根桥交换机（AS 交换机）的那一个下行接口上（做在 DS 交换机接口上），主备根桥之间的链路不需要配置

6、AS 交换机不能配置 ip route 0.0.0.0 0.0.0.0 网关地址 这样来配置网关，必须使用 ip default-gateway 网关地址 来配置网关，这样是和上面一样的效果，并且对于二层设备来说更规范

7、可以设置显示十进制的 tag（类似 IPv4 的格式），但是默认是不会显示为十进制的，使用全局命令开启：route-tag notation dotted-decimal。然后就可以再 （可以用 route-map 等命令指定十进制 tag），使用 show 命令查看时就能够查看到十进制了

```
IR2#show ip route 10.47.0.21
Routing entry for 10.47.0.21/32
  Known via "ospfv3 39", distance 110, metric 20
  Tag 66.66.66.66, type extern 2, forward metric 1
  Redistributing via eigrp 4839
  Advertised by eigrp 4839 metric 1000000 10 255 1 1500 route-map
  Last update from 10.47.1.30 on GigabitEthernet0/1, 00:00:27 ago
  Routing Descriptor Blocks:
  * 10.47.1.30, from 10.47.0.11, 00:00:27 ago, via GigabitEtherne
```

8、vlan 中如果有一个 vlan 是 blackhole（黑洞）vlan，那么建议不要在 trunk 接口中放行它，具体还要看场景，然后出来讨论

9、在 FHRP 中，如果题目要求定义 track，注意不要在从设备上定义和主设备一样的降低优先级，从设备可以有 track，但是当主设备降低优先级后，从设备的优先级要能够接管，交流赛时不小心配置了从设备和主设备降低一样的优先级，导致从设备无法接管

10、将 地址加入进 BGP vpnv4 之后，相应接口会自动配置一条 mpls bgp forwarding，必须有这条才能转发，并且这条可以被手动 no 掉，排错时需要注意

11、如果 bgp 添加了携带 community 标签的路由，那么一定要在所有 bgp 路由器中全局开启：ip bgp-community new-format
12、OSPF 对 Loopback 接口有特殊标识，哪怕它的地址掩码是/24 位的，通告它通告出去的掩码默认依旧是 /32 位，想要让它变回 /24，需要在该接口配置 ip ospf network point-to-point（主要还是看路由表、看题目要求）
13、dmvpn 如果是阶段三，eigrp 不需要设置 no next-self-hop，只有 hub 需要关闭水平分割
14、PBR 的 ip next hop 必须是接口直连下一跳，不能是 Loopback 地址，或者其他不是直连的地址
15、tp 最好加上 maxpoll 和 minpoll 参数（ntp master 不需要加）：
ntp server 10.255.0.1 source Loopback0 prefer maxpoll 4 minpoll 4 burst iburst

一、交换：

1、port-channel：

LACP —— IEEE 801.1AX/802.3ad

别名：
EtherChannel、Port-Channel、LAG

（1）LACP 热备份：

```
int port 1
 lacp max-bundle 1   //设置最大活动接口数量
 shutdown   //先关闭接口
int gig1/1
 lacp port-priority 1   //优先级越小越优先
int port 1
 no shutdown
```

为了防止生成树环路，需要在 port-channel 接口中调高 STP cost 值，让流量不通过 port-channel 接口

```
int port 1
 snapping-tree vlan 10 cost 100
```

（2）port-channel load-balance：

```
# 注意看题目要求，不要漏配了
port-channel load-balance ...
```

2、Trunk：

```
switchport trunk allowed vlan 101-103,888,999  //放行 VLAN
switchport trunk native vlan 888  //设置 Native VLAN
switchport nonegotiate  //关闭 DTP 协商
```

```
# 如果有的 vlan 名字叫做 BLACKHOLE，那么 trunk 链路上就不应该放行它
并且下面这些名称也不应该放行：
UNUSED  //未使用 vlan、
DEAD/DISCARD  //丢弃/废弃
RESERVED  //预留
PARKING/PARKING_LOT
```

3、VTP、Vlan：

（1）VTP：

注意，交换机重启后，VTPv3 Primary Server 会丢失，需要重新配置为 Primary Server

```
题目要求禁用 VTP，就不要开启 VTPv3

如果题目要求禁用 VTPv3，就要开启 VTPv3 并同时禁用这三个：
vtp domain WSC2026
vtp version 3
vtp mode off vlan
vtp mode off mst
vtp mode off unknow
```

•在指定接口关闭 VTP：

```
inter g0/1
 no vtp
```

•VTP Pruning：

```
vtp pruning
# 这个有风险，它可能导致二层不通，如果赛题要求配置，要考虑是否配置它。
```

4、STP：

```
STP/PVST+（默认） —— IEEE 802.1D
```

RSTP —— IEEE 802.1w
MSTP —— IEEE 802.1s

（1）控制链路选举：

1.1、修改 cost 值：

先比较 cost 值，再比较 port-priority

```
## 需要在本端出方向更改：
## 用于控制那条路径更优先（值越小越优先）
int g0/0
 spanning-tree vlan 10 cost 100
```

1.2、修改 接口优先级（Port-Priority）：

当 cost 值相同时才比较 port-priority

```
# 值为 16 的倍数，最大 144：
## 需要在对端入方向更改：
int f0/1
 spanning-tree vlan 10 port-priority  144
```

（2）root guard、bpduguard：

```
# bpduguard、bpdufilter、portfast 要注意题目说的是单个接口，还是全局配置，例如可能这样表述：
Configure Gi0/10 as an edge port and protect it from received BPDUs   //单个接口
Enable PortFast on access ports   //全局
Protect edge ports against the connection of unauthorized switches   //全局

# root guard：
spanning-tree root guard 要做在 DS 交换机 连接 其他非根桥交换机（AS 交换机）的那一个下行接口上（做在 DS 交换机接口上），主备根桥之间的链路不需要配置

# loop guard：
配置在阻塞接口上
```

5、port-security：

```
# 三种模式：
1、Protect：丢弃违规流量，不告警，不记录日志
2、restrict：丢弃违规流量，告警并记录日志
```

3、shutdown（默认）：逻辑关闭接口，并记录日志

```
# port-security Sticky 老化定时器：
## 默认永不老化：
int g0/0
 switchport port-security aging time 10   //设置老化时间 10 分钟
 switchport port-security aging type [absolute/inactivity]   //两种类型
  absolute   //直接开始计时，10 分钟一道立马删除
  inactivity   //当没有流量时才开始计时
```

6、PVLAN（二层隔离）

```
# 同时处于 protected 的接口不能进行二层通信
switchport protected
```

7、DHCP、DHCP Snooping：

```
# 常用 Option：
Option 3: 网关
Option 6：DNS Server
Option 15：DNS 域名
Option 42：NTP
```

```
# DHCP Snooping:
ip dhcp snooping
ip dhcp snooping vlan 10
ip dhcp snooping information option
# 然后在上游接口 trust：
int g0/0
 ip dhcp snooping trust
```

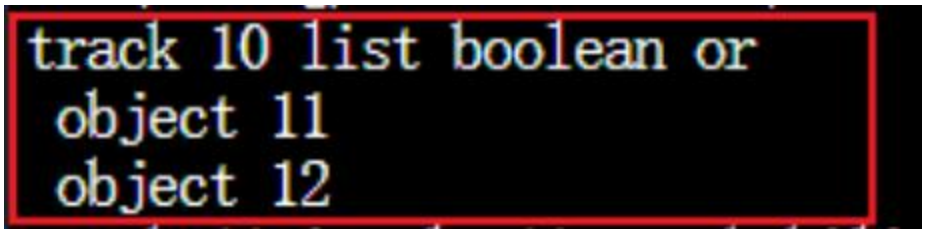
8、DAI：

```
# 动态 DAI 需要依赖 DHCP Snooping 数据库
# 静态 DAI 需要手动配置 arp acl
## 静态 DAI 示例：
arp access-list ARP
 permit ip host 192.168.10.10 mac host 5254.0007.4792
ip arp inspection filter ARP vlan 10
ip arp inspection vlan 10
interface range g0/0-1
```

```
ip arp inspection trust
```

9、FHRP：

可以用 track object 同时匹配多个 track，当它同时 down 时才切换：



（1）HSRP：

```
注意事项：IPv6 需要开启：standby use-bia
```

```
# 自定义虚拟 MAC 地址：固定格式，0000.0C07.ACXX
使用组号来控制 MAC 地址 XX 的值，它的值是 十六进制
HSRP 组号：1 ——> 01 ——> 0000.0C07.AC01
HSRP 组号：10 ——> 0A ——> 0000.0C07.AC0A
HSRP 组号：16 ——> 10 ——> 0000.0C07.AC10
```

（2）VRRP：

```
# 自定义虚拟 MAC 地址：固定格式，0000.5E00.01XX
vrrp 组号：1 ——> 01 ——> 0000.5E00.0101
vrrp 组号：10 ——> 0A ——> 0000.5E00.010A
vrrp 组号：16 ——> 10 ——> 0000.5E00.0110
vrrp 组号：255 ——> FF ——> 0000.5E00.01FF
```

用 calc 算就行了

```
fhrp version vrrp v3   //开启 VRRPv3
```

（3）GLBP：

与 HSRP、VRRP 不同，它这里两台交换机能够同时转发流量

```
# 负载均衡算法：
glbp 10 load-balancing ?
 round-robin   //轮询
```

```
host-dependent   //同一主机（MAC），尽量调度给同一个设备
weighted   //根据 GLBP 定义的权重来调度
```

10、ip default-gateway：

二层交换机要用 ip default-gateway 来配置网关

二、路由：

1、OSPF：

•OSPF 身份认证：

```
# 身份认证：
router ospf 1
 area 10 authentication message-digest   //表示这个区域的接口都需要身份认证
int g0/0
 ip ospf authentication message-digest
 ip ospf message-digest-key 1 md5 Skills39!

# 只有 OSPFv3 支持 ipsec 认证：
## 可以自己定义密码字符：md5 32 位十六进制字符；sha1 40 位十六进制字符
int g0/0
 ospfv3 1 ipv6 area 0 ospfv3 authentication ipsec spi 256 sha1 0
0987654321098765432109876543210987654321
```

•Stub 区域：

```
area 1 stub   //过滤 4、5 类 LAS，不过滤 3 类 LSA，不允许重分发
area 1 stub no-summary   //过滤 3、4、5 类 LSA，不允许重分发
area 1 nssa   //过滤 4、5 类 LAS，不过滤 3 类 LSA，允许重分发
area 1 nssa no-summary   //过滤 3、4、5 类 LSA，允许重分发

# 修改 stub 区域 下发的默认路由的 metric 值：
area 1  default-cost [要设置的 metric 值]
area 1 default-cost 0
# 设置完成后，它下发的 metric 值会默认 + 1 （例如，设置了 area 1 default-cost 0，那么下发的路由的 metric 值就是 1 ）
```

•网络类型：

```
# 在 DMVPN 场景中，网络类型建议设置为 point-to-multipoint
```

```
ip ospf network broadcast  //选举 DR/BDR，二层交换网络
ip ospf network non-broadcast  //选举 DR/BDR，需要 neighbor 手动建立邻居（不常用）
ip ospf network point-to-point  //不选举 DR/BDR，两台路由器直连
ip ospf network point-to-multipoint  //不选举 DR/BDR，Hub-Spoke 拓扑，Hub 同时连多个 Spoke
ip ospf network point-to-multipoint non-broadcast  //不选举 DR/BDR（不常用）
LOOPBACK  //Loopback 接口默认的网络类型，默认将地址以/32 位通告，除非手动配置网络类型为 point-to-point
```

•路由汇总：

```
# 汇总内部路由（区域间汇总）：
router ospf 1
 area 0 range 192.168.0.0 255.255.0.0
 # 被汇总的路由在什么区域，这个 area 就写哪个区域

# 汇总外部路由（需要在 ASBR 上）：
router ospf 1
 summary-address 192.168.0.0 255.255.0.0
```

•开启 bfd 检测：

```
int g0/0  //先在接口配置
 brd interval 50 min_rx 50 multiplier 3
router ospf 1
 bfd all-interfaces
 # 或者只对单个接口开启：bfd interface g0/0
```

2、EIGRP：

•身份认证：

```
interface g0/0
 ip authentication mode eigrp 2024 md5
 ip authentication key-chain eigrp 2024 KEY

# 只有 EIGRP 命名模式才能配置 hash 256 认证：
# 配置 key chain:
```

```
key chain KEY
 key 1
  key-string Skills39
  cryptographic-algorithm hmac-sha-256
router eigrp lyon2024
 address-family ipv4 autonomous-system 2024
  af-interface g0/0
   authentication mode hmac-sha-256
   authentication key-chain KEY
```

•路由汇总：

```
# 在接口汇总：
int tunn 0
 ip summary-address eigrp 3947 192.168.0.0 255.255.0.0
```

•重分发 Metric 值：

```
# 建议 metric 值：
metric 1000000 10 255 1 1500
```

•k 值：

```
K1  //带宽 bandwidth
K2  //负载 load
K3  //延迟 delay
K4  //可靠性 reliability
K5  //可靠性修正参数 reliability（决定 k4 是否参与，如果 k4 为 1，k5 为 0，那么 k4 还是不参与计算）

默认：
router eigrp 100
 metric weights 0 1 0 1 0 0  //第一个固定是 0
```

•Stub 区域：

```
router eigrp 1
 eigrp stub ?
# 参数（多个参数可以组合使用）：
 connected  只发送直连路由给邻居
 leak-map  通过 route-map 来手动匹配路由，只发送该 route-map 匹配的路由给邻居
 receive-only  只接收更新，不发送任何更新
 redistributed  只发送 重分发的外部路由 给邻居
 static  只发送本地静态路由
```

```
summary  只发送汇总路由
```

•开启 bfd 检测：

```
int g0/0  //先在接口配置
 brd interval 50 min_rx 50 multiplier 3
router eigrp 100
 bfd all-interfaces
 # 或者指定接口：bfd interface g0/0
```

3、BGP：

•BGP 反射器：

```
根据场景用反射器解决 BGP 的水平分割问题
```

•BGP Community

```
# 如果环境中为路由设置了 community 值，必须要配置：
1、ip bgp-community new-format  //全局开启。能够显示完整的 community，例如：65000:100；如果不开就只能这样显示：49320912
2、在邻居中配置 send-community
# Community 参数：
route-map BGP permit 10
 match ip address 1
 set community no-export

# 参数：
internet  //（收到这条路由的路由器）允许通告给所有 BGP 邻居
local-AS  //（收到这条路由的路由器）不发出本地 AS（不发给 eBGP 邻居）（正常情况下与 no-export 一样），用于 bgp 联邦环境，只会在 bgp 联邦中的其中一个 AS 中传递，不会传递给联邦其他 AS
no-advertise  //（收到这条路由的路由器）不发将这条送给任何邻居（iBGP/eBGP）
no-export  //（收到这条路由的路由器）可以发给 iBGP 邻居，不能发给 eBGP 邻居
gshut  //告诉其他路由器，这条路由正在维护，可以配合路由策略降低这条路由的优先级
none  //删除路由的特殊属性，只保留 community
```

•身份认证：

```
router bgp 64512
 neighbor 98.76.54.1 remote-as 65002
 neighbor 98.76.54.1 password Pa$$worD
```

•路由汇总：

```
router bgp 70000
 aggregate-address 17.1.0.0 255.255.0.0 summary-only

# 参数：
1、是否生成汇总
route-map  //（定义是否汇总路由）调用 route-map，只在 route-map 匹配的指定明细路由存在时才进行汇总
advertise-map  //（定义是否向外通告）调用 route-map，总是会生成路由汇总，但是只有当指定条件匹配时才对外发布路由，否则该汇总路由只会在本地存在

2、是否发布明细路由
summary-only  //仅发布汇总路由，不同时发布明细路由
suppress-map  //调用 route-map，类似于 eigrp 的 leak-map（但不完全相同，leak-map 是匹配到的明细会通告，这个是匹配到的明细会被抑制，所有没匹配到的明细都会被通告），在发布汇总路由的同时，抑制所有匹配到的明细，剩余没有被匹配到的明细，将被放行

3、汇总路由携带那些 BGP 属性（Local-Preference、metric、weight）
attribute-map  //调用 route-map，为该汇总路由设置 BGP 属性（Local-Preference、metric、weight）（不能指定 match ip.. 参数匹配地址前缀）

4、AS_PATH 如何表现
as-set  //汇总所有明细路由的 AS 到汇总路由，如果不加，汇总路由只会携带一个本地的 AS 号[不加 as-set，AS_PATH: 70000；加了 as-set，AS_PATH: {65001 65002} 70000]
as-confed-set  //只在 BGP 联邦的场景下生效，决定是否保留 BGP 联邦内部的 AS-PATH，加了 as-confed-set，就会在汇总路由传递时，保留联邦内部的 AS-PATH，当这条汇总路由传出联邦时，携带的联邦私有 AS 号会被隐藏，只显示联邦公用 AS 号
```

•Template、peer-group：

```
# Template：
router bgp 70000
 template peer-session SESSION
  remote-as 70000
  update-source Loopback0
 template peer-policy POLICY
  route-reflector-client
```

```
 neighbor 2.2.2.2 inherit peer-session SESSION
 neighbor 2.2.2.2 inherit peer-policy POLICY
 neighbor 4.4.4.4 inherit peer-session SESSION
 neighbor 4.4.4.4 inherit peer-policy POLICY

# peer-group：
 router bgp 100
 address-family ipv4
  neighbor INTERNAL-IPv4 route-reflector-client
  neighbor INTERNAL-IPv4 next-hop-self all

  neighbor 150.1.1.1 peer-group INTERNAL-IPv4
  neighbor 150.1.2.2 peer-group INTERNAL-IPv4
  neighbor 150.1.3.3 peer-group INTERNAL-IPv4
  neighbor 150.1.5.5 peer-group INTERNAL-IPv4
```

•开启负载均衡链路：

```
# 默认只允许一条最优路径
router bgp 2022
 maximum-paths eibgp [数字]  //允许 eBGP 多路径，后面的数字定义允许的负载均衡路由的数量（1 ~ 32）
 maximum-paths ibgp [数字]  //允许 ibgp 多路径，后面的数字定义允许的负载均衡路由的数量（1 ~ 32）
```

•条件路由注入：

```
# 当某个条件生效时，才将路由导入路由表（当某条路由存在时）bgp 专属功能：
ip prefix-list EXIST seq 5 permit 17.1.0.0/16   //定义要匹配的路由
ip prefix-list SOURCE seq 5 permit 100.3.7.3/32   //定义要匹配的路由的来源（哪条 bgp 邻居）
ip prefix-list INJECT seq 5 permit 17.1.1.0/24   //定义要添加的路由

route-map INJECT permit 10
 set ip address prefix-list INJECT
 # set ip next-hop ...   //（可选，手动设置路由下一跳）
route-map EXIST permit 10
 match ip address prefix-list EXIST
 match ip route-source prefix-list SOURCE

router bgp 65534
 bgp inject-map INJECT exist-map EXIST copy-attributes   //copy-attributes 属性作用：复制原
```

始路由的所有属性，例如 next-hop、as_path、community（一般都建议加上这个属性，如果不加，并且也不在 route-map 中用 set 自定义属性，那么这条路由就会变成本地产生的路由）

•单独针对邻居进行操作：

```
# 单独设置邻居路由的 weight：
router bgp 65533
 neighbor 100.22.23.23 weight 100

# 单独对邻居下发默认路由（本地不需要存在默认路由）：
# iBGP/eBGP 都能生效：
router bgp 2120
 neighbor 21.1.1.1 default-originate   //对指定邻居下发一条默认路由（不需要本地存在默认路由条目）

## 一般可以同时再配置一条过滤（主要看题目要求，并且它不会过滤掉这条下发的默认路由）：
access-list 1 deny any
route-map R21 permit 10
 match ip address 1
router bgp 2120
 neighbor 21.1.1.1 route-map R21 out
```

4、IPv6：

•SLAAC：

```
别名：
Stateless Address Autoconfiguration
SLAAC
IPv6 autonomous address configuration
RA-based IPv6 addressing
```

```
# 禁用 SLAAC：
## 这个命令同时也会导致不会下发默认路由：
int g0/1
 ipv6 nd ra suppress

## 这个命令会只禁用 SLAAC 地址生成，但是仍然会下发默认路由（适用于 DHCPv6）：
```

```
int g0/1
 ipv6 nd prefix default 60480 60480 no-autoconfig
```

•DHCPv6：

注意，DHCPv6 无法下发默认路由，默认路由只能由 RA 下发，所以关闭 slaac 只能用 no-autoconfig 的方法

```
# 配置 DHCPv6 后，要在接口开启 m 和 o 选项
int g0/1
 ipv6 nd managed-config-flag
 ipv6 nd other-config-flag

# DHCPv6：
ipv6 dhcp pool DHCP
 address prefix 2001:DB8:2026:102::/64
 dns-server 2001:DB8:2026:102::1
 domain-name wsc2026.net
## 在接口应用：
int g0/0
 ipv6 dhcp server DHCP
```

5、路由控制 方法：

•tag：

可以设置显示十进制的 tag（类似 IPv4 的格式），但是默认是不会显示为十进制的，使用全局命令开启：route-tag notation dotted-decimal。然后就可以再 （可以用 route-map 等命令指定十进制 tag），使用 show 命令查看时就能够查看到十进制了

（1）EIGRP：

1.1、修改接口 delay（不推荐）：

接口默认 delay：10
delay 越小，metric 值越小

```
# 修改出接口的 delay（路由从哪边出去，就改哪边的 delay）
interface g0/0
 delay 1
```

1.2、offset-list：

in 方向：对邻居传入的路由增加跳数/度量值
out 方向：对从本地传给邻居的路由增加跳数/度量值

```
access-list 1 permit 1.1.1.1 /32
router eigrp 100
 # 第一个值指定 0 可以匹配所有路由
 # 第二个值表示要增加的路由跳数/度量值
 # 最后的接口可以不加，加了就只对单个接口生效，不加就对所有接口生效
 offset-list 1 out 1 g0/1
```

1.3、在重分发时调用 route-map 来修改 metric 值

1.4、配置 非等价负载均衡路由：

```
# 默认最多存在 4 条 等价/非等价负载均衡路由
router eigrp 1
 variance 2   //备份路由不超过最优路由两倍就能作为非等价负载均衡路由
 maximum-paths 2   //定义最多允许存在多少条 负载均衡路由
```

（2）OSPF：

2.1、修改接口 cost 值（最推荐）：

cost 默认值为 1（值越低越优先）
改 cost 值会影响出去的路由，例如 R1 去往 R2 有 G0/0、G0/1 两个接口，将 G0/0 接口的 cost 值设置为 10，那么流量就不会走 G0/0 了，会优先走 G0/1

```
# OSPFv2：
interface GigabitEthernet0/0
 ip ospf cost 10

# OSPFv3：
interface GigabitEthernet0/0
 ospfv3 cost 10
```

2.2、修改 OSPF 参考带宽，让它自动计算 cost 值：

必须在 OSPF 所有设备（哪怕不同 area）中配置，否则会导致选路问题
其实不会影响到路由选路，会自动计算 cost，用 下面定义的 bandwidth 除以 真实带宽 就等于 cost 值（cisco 中真实带宽默认都是 1000）

```
router ospf 1
 auto-cost reference-bandwidth 100000      // 这 样 cost 会 自 动 计 算 为 100 （ 因 为
```

100000/1000=100）
#auto-cost reference-bandwidth 10000 // 这样 cost 会自动计算为 10（因为 10000/1000=10）

2.3、在重分发时调用 route-map 来修改 metric 值、将路由类型改为 E1 让它通过链路自动叠加 cost。

2.4、max-metric 将本机传出去的所有路由设置为最大 cost 值：

让本机继续保持 OSPF 邻居关系，但是把经过本机的路由，metric（cost）设置为最大值，让流量尽量先不通过该路由器
如果本机作为区域 ABR，还会影响区域间的路由，让它优先不走本机

比赛如果要考，大概率只会考 on-startup 这个参数，因为其他参数都能用 route-map 做到一样的效果

一直将 cost 设置为最大值：
默认只影响经过的路由，本机的直连网段（没有建立 ospf 邻居的直连网段），这个网段的路由也不会影响，同时默认也不会影响 外部路由、汇总路由。这些都要额外指定参数
router ospf 1
max-metric router-lsa
所以假设题目要求将该路由器设置为备份路由器，那么一般这样配置：
max-metric router-lsa summary-lsa external-lsa //并且还可以手动指定将这些路由改为什么 cost 值，而不是必须最大值

将本机产生的 外部路由、汇总路由、本地直连路由 的 cost 也修改：
默认不会修改 汇总路由、外部路由、直连路由（本机独有，没有其他 OSPF 邻居的路由）的 cost 值
可以组合使用，也可以单独使用：
router ospf 1
max-metric router-lsa summary-lsa external-lsa
max-metric router-lsa include-stub //同时也影响本机独有的直连路由（无 ospf 邻居）

路由器启动后的多少秒内，不转发流量（将 cost 值设置为最大值）：
router ospf 1
max-metric router-lsa on-startup 10

等待 BGP 路由收敛后，再将路由 cost 值恢复到正常（当路由器启动后）：
假设本机路由器同时运行 ospf 和 bgp，当机器重启后，ospf 启动和获取路由会比 BGP 更快，此时如果有流量来到本机，但是 bgp 还没有收敛完成（例如 bgp 默认路由还没有），那么该流量就会被丢弃

（如果 bgp 一直不收敛，最多等待 600 秒）
router ospf 1
 max-metric router-lsa on-startup wait-for-bgp

(3) BGP:

BGP 按下面的顺序来选择最优路径（当值相同时才选择下一项）：
1、最高 Weight 值
2、最高 LOCAL_PREF 值
3、是否为本地发起路由（下一跳为 0.0.0.0 的路由）
4、最短 AS_PATH（使用 bgp bestpath as-path ignore 可以忽略对 AS_PATH 的比较）（使用 ip as-path acl 可以手动增加 AS_PATH）
5、最优 Origin 类型（IGP 优于 EGP 优于 Incomplete[重分发]）
6、最小 MED 值
7、eBGP 优于 iBGP
8、最优 IGP metric 值
9、负载均衡（如果开启），如果上面的值全都相同（包括 AS-PATH），就会开启负载均衡（最大支持 6 条，默认为 1 条），否则，继续比较下一个属性
10、如果下一跳都为 eBGP，则选择最早学习到的路由（存在时间最长的路由）
11、最低 Router-ID 下一跳
12、最短 cluster list（只在 reflector RR 反射器的环境存在，功能如同 AS_PATH）
13、最小下一跳的邻居地址

1、修改 Local Preference：
仅影响当前 iBGP，仅在 iBGP 范围内传播
值越大越优先，默认值 100
ip prefix-list TO-INTERNET permit 0.0.0.0/0
route-map SET-LOCALPREF permit 10
 match ip address prefix-list TO-INTERNET
 set local-preference 200
router bgp 65001
 neighbor 12.1.1.1 route-map SET-LOCALPREF in

2、weight
仅本地有效，不会传给邻居
值越大越优先，默认值 0（本地生成的路由 32768）
route-map SET-WEIGHT permit 10
 set weight 500
router bgp 65001
 neighbor 12.1.1.1 route-map SET-WEIGHT in
或者直接设置邻居 weight 值：
router bgp 65001
 neighbor 12.1.1.1 weight 500 //从该邻居学习到的所以路由 weight 都设置为 500

3、修改 AS-PATH，影响 eBGP 路径
增加本地的 AS-PATH 长度，就能影响 eBGP 优先走另一台路由器

```
ip prefix-list MY-PREFIX permit 192.168.10.0/24
route-map PREPEND permit 10
 match ip address prefix-list MY-PREFIX
 set as-path prepend 65001 65001 65001
router bgp 65001
 neighbor 12.1.1.1 route-map PREPEND out
```

4、community
用 community 为路由打上标签，然后针对这个 community 标签进行控制、过滤

```
ip prefix-list MY-PREFIX permit 192.168.10.0/24
route-map SET-COMM permit 10
 match ip address prefix-list MY-PREFIX
 set community 65000:100 additive
router bgp 65001
 neighbor 12.1.1.1 send-community
 neighbor 12.1.1.1 route-map SET-COMM out
```

5、MED：
可以影响 eBGP 和 iBGP
越小越优先，默认值 0

```
ip prefix-list MY-PREFIX permit 192.168.10.0/24
route-map SET-MED permit 10
 match ip address prefix-list MY-PREFIX
 set metric 50
router bgp 65001
 neighbor 12.1.1.1 route-map SET-MED out
```

(4) BGP 额外参数：

```
# 选路时忽略 AS_PATH：
router bgp 65001
 bgp bestpath as-path ignore
```

让相同 AS 之间比较 MED 更加稳定：
优化 AS 之间的 MED 比较，让相同 AS 的进入同一个组进行比较，选出最优的那条（这条是优化选项，可以常驻配置，不会引发问题）
因为默认选择比较 MED 时，如果有其他 AS 参与，可能不会选择到最优路径

```
router bgp 65001
 bgp deterministic-med

# 不管来自哪个 AS，都强制比较 MED：
不管来自哪个 AS，都强制比较 MED：
router bgp 65000
 bgp always-compare-med
```

（5）不同协议间选路：

•默认选路：

路由来源	默认 AD
Connected 直连	0
Static 静态	1
EIGRP Summary	5
eBGP	20
EIGRP Internal	90
OSPF	110
EIGRP External	170
iBGP	200

```
# 默认是这样选路的：
直连 > 静态 > EIGRP 汇总 > eBGP > EIGRP 内部 > OSPF > EIGRP 外部 > iBGP
```

•降低 AD 值的命令：

```
# EIGRP：
router eigrp 1
 distance eigrp 90 100   //设置内部路由 AD 值为 90（不变），外部路由设置为 100

# OSPF：
router ospf 1
 distance ospf intra-area 100 inter-area 100 external 121
 ## intra-area：区域内
 ## inter-area：区域间
 ## external：外部

# BGP：
```

```
router bgp 65000
 distance bgp 20 100 200
 # 第一个值 20：eBGP 路由 AD 值
 # 第二个值 100：iBGP 路由 AD 值
 # 第三个值 200：本地产生的 BGP 路由的 AD 值
```

(6) PBR:

6.1、在 源接口应用，控制通过的路由:

意，要用扩展 ACL 匹配目标路由，而不是源路由

```
# 配置 ACL 匹配路由：
## 注意，要用扩展 ACL 匹配目标路由：
ip access-list extended PBR
 permit ip any 192.168.1.0 0.0.0.255

# 配置 route-map 设置下一跳：
route-map PBR permit 10
 match ip address PBR
 set ip next-hop 1.1.1.1

# 在 源接口应用 PBR：（该路由从哪个接口进入，就在哪个接口应用）
interface g0/0
 ip policy route-map PBR
```

6.2、在 本地应用，控制始发于本地的路由:

```
# 配置 ACL 匹配路由：
ip access-list extended PBR
 permit ip any 192.168.1.0 0.0.0.255

# 配置 route-map 设置下一跳：
route-map PBR permit 10
 match ip address PBR / match interface Loopback 0   //这里使用 match interface 会好一点
 set ip next-hop 1.1.1.1

# 应用 PBR（控制始发于本地的路由）
ip local policy route-map PBR
```

6、路由过滤：

（1）office-list：

可使用 office-list 的路由协议：RIP、EIGRP
不能使用 office-list 的路由协议：ospf、BGP

1.1、RIP / EIGRP：

in 方向：对邻居传入的路由增加跳数/度量值
out 方向：对从本地传给邻居的路由增加跳数/度量值

```
access-list 1 permit 1.1.1.1 /32
router eigrp 100
 # 第一个值指定 0 可以匹配所有路由
 # 第二个值表示要增加的路由跳数/度量值
 # 最后的接口可以不加，加了就只对单个接口生效，不加就对所有接口生效
 offset-list 1 out 1 g0/1
```

（2）OSPF 专用：

```
# ip ospf prefix-suppression
## 作用，该直连接口仅建立 OSPF 邻居，但是不将该接口的 OSPF 网段通告进 OSPF 进程中：
## 可以用 distribute-list + filter-list 实现一样的效果，但是不如这一条命令方便快捷
int g0/0   //两端邻居都配置
 ip ospf prefix-suppression
```

（3）distribute-list：

可使用 distribute-list 的路由协议：RIP、EIGRP、ospf、BGP
但是 BGP 不建议使用 distribute-list，直接使用 route-map 更好

3.1、RIP/EIGRP：

in 方向：过滤邻居传入的路由
out 方向：过滤从本地传给邻居的路由

```
ip prefix-list R1_Loop0 seq 5 deny 1.1.1.1/32
ip prefix-list R1_Loop0 seq 10 permit 0.0.0.0/0 le 32

router eigrp 100
```

```
# 最后的接口可以不加，加了就只过滤单个接口的路由，不加就过滤所有接口的路由
distribute-list prefix R1_Loop0 in g0/2
```

3.2、OSPF:

注意：OSPF 只能过滤 in 方向的路由，并且只影响路由进入本地路由表，不影响该路由发给其他邻居

```
ip prefix-list R1_Loop0 seq 5 deny 1.1.1.1/32
ip prefix-list R1_Loop0 seq 10 permit 0.0.0.0/0 le 32

router ospf 100
# 最后的接口可以不加，加了就只过滤单个接口的路由，不加就过滤所有接口的路由
distribute-list prefix R1_Loop0 in g0/2
```

distribute-list out 方向在 OSPF 其实也能生效，但是只能过滤本地重分发进 ospf 进程的路由，例如重分发 eigrp，但是这样做多此一举，在重分发时调用 route-map 过滤还更简单

3.3、BGP（比较鸡肋）：

BGP 也支持 distribute-list 过滤，但是跟 ospf 的 out 方向一样鸡肋，还不如直接使用 route-map

```
router bgp 100
 neighbor 12.1.1.2 distribute-list 10 in
 neighbor 12.1.1.2 distribute-list 20 out

# 还不如使用 route-map：
neighbor 12.1.1.2 route-map RM-IN in
neighbor 12.1.1.2 route-map RM-OUT out
```

(4) filter-list:

```
可使用 filter-list 的路由协议：ospf、BGP
不可使用 filter-list 的路由协议 RIP、EIGRP、

BGP filter-list：用 AS-PATH ACL 过滤 BGP 路由。
OSPF area filter-list：在 ABR 上过滤区域间路由，也就是 3 类 LSA。
```

4.1、OSPF（ABR 上配置）：

只能在 ABR 上配置，只能过滤三类 LSA（区域间路由：O IA 路由），一类、二类 LSA（区域内路由）过滤不了，也无法过滤五类、7 类 LSA（重分发路由），当然，5 类、7 类路由可

以在重分发时直接 route-map 过滤
只能使用 prefix-list 进行匹配
area 1 ... in：过滤从其他区域传入 area 1 的路由：3 类 LSA
area 1 ... out：过滤从 area 1 出去的路由（过滤从 area 1 传递到其他区域的路由：3 类 LSA）

```
ip prefix-list R1_Loop0 seq 5 deny 1.1.1.1/32
ip prefix-list R1_Loop0 seq 10 permit 0.0.0.0/0 le 32

router ospf 1
 area 1 filter-list prefix R1_Loop0 in
```

4.2、BGP（配置 as-path acl 使用）：

只能通过 AS-PATH 属性过滤

```
# 定义 as-path list（这个需要使用正则表达式匹配）：
## 匹配本地始发的路由（next-hop 为：0.0.0.0 的路由）：
ip as-path access-list 10 permit ^$

## 匹配所有 BGP 路由：
ip as-path access-list 10 permit .*

## 只接收 AS 200 的路由（匹配 AS PATH 中只有 AS 200 的路由）：
ip as-path access-list 10 permit ^200$
ip as-path access-list 10 deny .*

## 禁止将 AS-PATH 包含 AS 100 的路由通告（过滤经过 AS 100 的路由）：
ip as-path access-list 10 deny _100_
ip as-path access-list 10 permit .*

## 匹配以 AS 100 结尾的路由（匹配始发于 AS 100 的路由）：
ip as-path access-list 10 permit _100$
ip as-path access-list 10 deny .*

## 匹配以 AS 100 开头的路由：
ip as-path access-list 10 permit ^100_
ip as-path access-list 10 deny .*

### as-path access-list 默认末尾有一条 deny .* 拒绝所有的规则
# AS-PATH 的方向：
左边 = 离你最近的 AS
右边 = 路由的始发 AS / Origin AS
# 例如：
100 200 300
```

这条路由经过 AS 300 始发，经过 AS 200，再经过 AS 100，最后传到这里

需要注意如果开了反射器之类的、有多路径之类的，把最优路由拒绝了该路由是否会通过其他邻居的次优路径再传进来

neighbor ... filter-list 10 in：邻居传入路由时，将路由用 as-path access-list 过滤一遍

neighbor ... filter-list 10 out：传路由给邻居前，将路由用 as-path access-list 过滤一遍

```
# 允许始发于 AS 200 的路由，过滤经过 AS 200 的路由：
ip as-path access-list 10 permit _200$
ip as-path access-list 10 deny _200_
ip as-path access-list 10 permit .*

router bgp 400
 neighbor 100.2.4.2 filter-list 10 in
 neighbor 100.3.4.3 filter-list 10 in
```

```
R4(config-router)#do show bgp
BGP table version is 1, local router ID is 100.4.5.4
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
              t secondary path,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *    1.1.1.1/32       100.3.4.3                              0 300 100 i
 *    100.1.2.0/24     100.3.4.3                              0 300 100 i
 *                     100.2.4.2                0             0 200 i
 *    100.1.3.0/24     100.3.4.3                0             0 300 i
 *    100.3.4.0/24     100.3.4.3                0             0 300 i
R4(config-router)#
```

（5）database-filter：

```
可使用 database-filter 的路由协议：ospf
不可使用 datebase-filter 的路由协议 RIP、EIGRP、BGP


database-filter 无法直接过滤指定路由，但是可以禁止向指定接口发送（泛洪）SLA（相当于过滤所有 OSPF 路由）
与被动接口的区别：
database-filter：不发送 LSA，仍然能够发送 Hello 包建立邻居
passive-interface：不发送 Hello，不建立邻居，自然也不会发送 LSA
## 根据试题要求判断要选择哪个进行配置
```

5.1、OSPF：

```
## 接口配置：
# 不通过 G0/0 接口向外发送 LSA（与 G0/0 接口建立邻接关系的邻居将无法从本机获取到任
```

```
何 OSPF 路由，但是这个邻居它自身的 OSPF 路由还是能够正常传递给本机）
interface GigabitEthernet0/0
 ip ospf database-filter all out

## 单独针对邻居配置（指定邻居 router-id）：
router ospf 1
 neighbor 5.5.5.5 database-filter all out
```

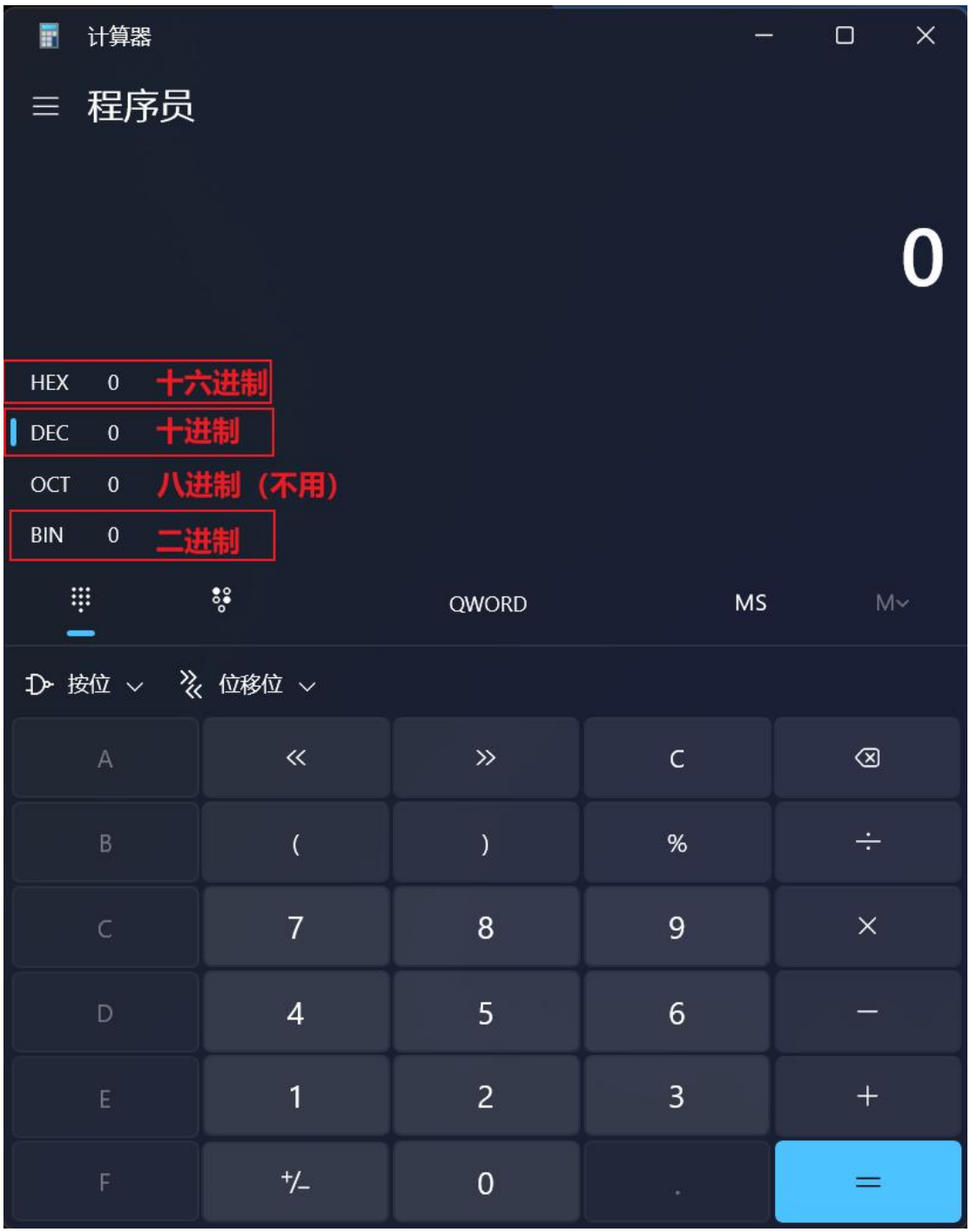
(6) PBR:

```
# 在 源接口应用，控制通过的路由
## 配置 ACL 匹配路由：
access-list 10 permit 192.168.1.0 0.0.0.255
## 配置 route-map 设置下一跳：
route-map PBR permit 10
 match ip address 10
 set ip next-hop 1.1.1.1
# 在 源接口应用 PBR：（该路由从哪个接口进入，就在哪个接口应用）
interface g0/0
 ip policy route-map PBR

# 在 本地应用，控制始发于本地的路由：
## 配置 ACL 匹配路由：
access-list 10 permit 192.168.1.0 0.0.0.255
## 配置 route-map 设置下一跳：
route-map PBR permit 10
 match ip address 10 / match interface Loopback 0   //这里使用 match interface 会好一点
 set ip next-hop 1.1.1.1
## 应用 PBR（控制始发于本地的路由）
ip local policy route-map PBR
```

7、计算路由汇总：

```
# 用 calc：
```



将地址转换为二进制，前面相同的位数就是网络号（剩下的位数补 0），
然后这一类的网络掩码 加上 这个相同数量的位数 就是掩码了（如果不足 8 比特就在前面补 0）
将最小的地址，和最大的地址计算为二进制：
第一组：
192.168.1.0/24
192.168.12.0/24
192.168.65.0/24
1：0 000 0001

65: 0 1000001
相同：0 0000000

计算 0000 0000 二进制地址转换为十进制，因此网络号就是 .0
掩码就是 16 + 1 = /17
汇总网段：192.168.0.0/17

第二组：
10.255.158.0
10.255.190.0
158：10 01 1110
190：10 11 1110
相同：10 000000

计算 1100 0000 二进制地址转换为十进制，因此网络号就是 .128
掩码就是 16 + 2 = /18
汇总网段：10.255.128.0/18

第三组：
172.16.11.0
172.16.19.0
11：000 0 1011
19：000 1 0011
相同：000 00000

计算 0000 0000 二进制地址转换为十进制，因此网络号就是 .0
掩码就是 16 + 3 = /19
汇总网段：172.16.0.0/19

IPv6 也是一样的思路，只不过用的是十六进制，一段是 32 比特（如果不足 32 比特就在前面补 0）
2001:DB8:1200::/41
2001:DB8:1234::/64
2001:DB8:1269::/64

1200：0001 0010 0 000 0000
1269：0001 0010 0 110 1001
相同：0001 0010 0 000 0000

计算 0001 0010 0 000 0000 二进制地址转换为 16 进制，因此网络号就是 .1200
掩码就是 32 + 9 = /41
汇总网段：2001:DB8:1200::/41

三、VPN

1、IPSec 加密：

1、proposal 阶段一加密一台路由器只能存在一个
2、申请证书时，最好手动指定证书私钥，否则会随便使用一个私钥
3、配置证书加密时，要关闭 http-url：no crypto ikev2 http-url cert

2、dmvpn：

1、阶段二 DMVPN，Hub 才需要开启 no ip next-hop-selp eigrp 100（分支不需要），并且只需要在 Hub 上关闭水平分割 no ip split-horizon eigrp 100
2、阶段三只需要在 Hub 上关闭水平分割 no ip split-horizon eigrp 100，并设置 ip nhrp redirect

```
# Tunnel 接口 MTU 建议设置为 1400：
int tunn 0
 ip mtu 1400
```

四、服务

1、NTP：

1、有时同步不上，需要用 source 指定源接口（一定要注意路由连通性）
2、最好加上 maxpoll、minpoll 和 prefer 参数（ntp master 不需要）：

```
ntp server 10.255.0.1 source Loopback0 prefer maxpoll 4 minpoll 4 burst iburst
```

2、日志显示信息：

给所有 debug 信息/日志信息 加上 本地日期时间 + 时区：

```
service timestamps debug datetime msec localtime show-timezone
service timestamps log datetime msec localtime show-timezone
```

3、SSH：

生成 SSH 私钥时，不要加 Label，同时要先设置域名，这样生成的 SSH 私钥名字就 hi 自动叫做 <主机名>.域名，例如：R1.wsc2026.net
如果题目要求配置了 ssh，并且做完还有多余时间，可以在 Cisco 客户端上设置 SSH 算法

4、NAT：

NAT 如果是地址池，那么就需要配置 ip route ... Null 0，并且地址池

5、NAT64：

（1）有状态：

•NAT64：

```
int g0/1
 nat64 enable
int g0/2
 nat64 enable

ipv6 access-list NAT64
 permit ipv6 2001:abcd::/64 any   /匹配能够通过 nat64 的 IPv6 地址
nat64 prefix stateful 6666:6666::/96
nat64 v4 pool V4 100.1.1.1 100.1.1.1   //定义转换后的 IPv4 地址（需要通告出去）
nat64 v6v4 list NAT64 pool V4 overload
```

•NAT46：

```
## 注意：这里的 NAT46 也是无法通过 NAT46 访问 NAT46 设备的 IPv6 接口
nat64 prefix stateful 6666:6666::/96   //用不到这个前缀，随意指定一个，让 NAT64 不报错
nat64 v6v4 static 2001:abcd::123 123.1.1.123   //将 2001:abcd::1 静态转换为 123.1.1.1
# 有多个 IP 就指定多条
```

```
nat64 v6v4 static 2001:abcd::5054:ff:fe0c:ee92 123.1.1.2
```

（2）无状态 NAT64：

可能不太会考，因为都是要求 IPv6 地址在 nat64 的前缀范围中

•静态 一对一转换：

```
nat64 prefix stateless 2001:ABCD::/96

 # 将外部 ipv4 地址引入 nat64 中，g0/0 表示该路由通过 g0/0 进行转发
nat64 route 136.136.136.136/32 GigabitEthernet0/1

# 配置路由（这里在 NAT64 机器和 Inside 机器上配置 OSPFv3）：
## NAT64：
ipv6 access-list NAT64
 permit ipv6 2001:ABCD::/96 any   //匹配 nat64 前缀并将它通告给 Inside
 permit ipv6 64:ff9b::/96 any   //匹配 nat64 自动生成的前缀，将其通告

route-map NAT64 permit 10
 match ipv6 address NAT64

router ospfv3 1
 router-id 1.1.1.2   //因为 ospfv3 必须要有一个 router-id 才能运行，我们随便写一个即可

 address-family ipv6 unicast
 redistribute static rotue-map NAT64   //仅重分发 nat64 前缀
```

•映射网段：

```
# NAT64 机器：
ipv6 un

int g0/1
 ipv6 enable
 nat64 enable   //需要在接口启用 nat64
 ipv6 address 2001:db8::fffe/64

int g0/0
 ip address 23.23.23.254/24 255.255.255.0
 nat64 enable
```

指定无状态 nat64 前缀（内部被转换的 ipv6 地址必须要包含在该前缀中）：
nat64 prefix stateless 6666:6666::/96

将外部 ipv4 地址引入 nat64 中，g0/0 表示该路由通过 g0/0 进行转发
nat64 route 200.1.1.0/24 g0/0

配置去往内部 ipv6 地址的路由，也可以配置动态路由，只要有路由能到达即可（注意对端需要有回程路由）
静态路由最好指定下一跳，否则有时无法生效
ipv6 route 6666:6666::CB01:0100/120 g0/0 2001:db8::1 // 注意：6666:6666::C801:0100/120 这个还是一个网段，因为最后一位为 0，且掩码不是 /128，如果是 6666:6666::C801:0100/128，那么就是一个 IP 地址
CB01:0100 是 200.1.1.0 转换为十六进制的值
当然如果不想转换为十六进制，也可以这样写：ipv6 route 6666:6666::200.1.1.0/120 g0/0 2001:db8::1 //（系统会自动将其转换为十六进制）

Inside 机器:
ipv6 route 6666:6666::/96 g0/0 2001:db8::254 //配置一条去往 NAT64 设备的 nat 前缀路由，否则路由无法通信（这条配置动态路由也可以）