Extra-Credit Project: RIP

- Due May 13th
- This is an extra credit project and uses an unofficial version of NW (created by Dr. Huang).
- Only Linux and Solaris versions are supported.
- If you wish to do this project, send me an email with title "request XXX RIP project," where XXX is either Linux or Solaris; I will send you an NW tar file in response.

- You implement the rip_receive() routing in file rip.cpp.

```c
rip_receive (RIP_segment* received_rip_segment,
            byte my_netnum, byte my_nodenum,
            byte ifacenum)
```

- this function is invoked when the router indicated by my_netnum and my_nodenum receives a distance (RIP) vector from one of its neighboring routers
- the purpose of the function is to update the local vector of the router according to the received vector
- ifacenum indicates the interface via which the incoming vector, received_rip_segment, is received
the format of RIP_segment is

```c
typedef struct {
    byte size;
    byte source_net;
    byte source_node; // always 1
    byte dest_net; // not used
    byte dest_node; // not used
    RIP_vector rip_vector; // the distance vector
} RIP_segment;
```

the structure of RIP_vector is

```c
typedef struct {
    byte number_of_entries;
    RIP_entry entries[MAX_NET+1];
} RIP_vector;
```

the structure of RIP_entry is

```c
typedef struct {
    byte destnet;
    byte ifacenum;
    byte cost; // # of hops
} RIP_entry;
```

further, the local distance (RIP) vector of the router
of network i is stored at rip_vectors[i]

each entry of the received vector is processed as
indicated by Rule 3 of distance vector routing (see
Page 8 of the routing slides).

- You also need to (re)implement function forward_iface() in
  file fwdopt.cpp based on the new routing table structure.
- To compile, cwkrip
- All projects must be individual efforts.