

Exercice 1 :

```
typedef struct wagon {  
    int reference;  
    int sieges;  
    int type;  
}WAGON;
```

```
typedef struct train {  
    WAGON donnee;  
    struct train* suivant;  
    struct train* precedent;  
}TRAIN;
```

// Question 1

```
TRAIN* deplacerFin(TRAIN* tete) {  
    if(tete==NULL) || tete->suivant==NULL) return tete;  
    else {  
        TRAIN *ref_tete=tete;  
        TRAIN *fin=tete;  
        while(fin->suivant!=NULL)  
            fin=fin->suivant;  
        tete=tete->suivant;  
        tete->precedent=NULL;  
        ref_tete->precedent=fin;  
        fin->suivant=ref_tete;  
        ref_tete->suivant=NULL;  
        return tete;
```

```
    }  
}
```

// Question 2

```
void afficherWagonVoyageur(TRAIN* tete){  
    if(tete==NULL) printf(" le train est vide");  
    TRAIN *courant=    tete;  
    while(courant!=NULL){  
        if(courant->donnee.type==1){  
            printf(" Refrence =%d",courant->donnee.reference);  
            printf(" Sièges =%d",courant->donnee.sieges);  
            courant=courant->suivant;  
        }  
        else {courant=courant->suivant;    }  
    }  
}
```

// Question 3

```
TRAIN* supprimerRestaurant(TRAIN* tete){  
    TRAIN*courant=tete;  
    TRAIN*precourant;  
    if(tete==NULL) return NULL;  
    if(tete->donnee.type==2){  
        tete=tete->suivant;  
        tete->precedent=NULL;  
        free(courant);  
        return tete;  
    }  
    else{
```

```

while((courant!=NULL)&& (courant->donnee.type!=2)){
    precourant=courant;
    courant=courant->suivant;
}
if(courant!=NULL){
    precourant->suivant=courant->suivant;
    courant->suivant->precedent=precourant;
    free(courant);
}
return tete;
}
}

```

// Question 4

```

TRAIN* supprimerTousRestaurants(TRAIN* tete){
    TRAIN*courant=tete;
    while(courant->suivant!=NULL){
        supprimerRestaurant(tete);
        courant=courant->suivant;
    }
    return tete;
}

```

Exercice 2 :

```

Struct Liste{
    int valeur ;
    struct Liste *suivant ;
};

```

// Question 1

```

void decoupe (struct Liste *L , struct Liste **L1 , struct Liste **L2){

    struct Liste *courant;

    *L1=L;

    if(L!=NULL) *L2=L->suivant;
    else *L2=NULL;

    while(L!=NULL){

        courant=L->suivant;

        if(courant!=NULL){

            L->suivant=courant->suivant;

            L=courant->suivant;

            if(courant->suivant!=NULL)

                courant->suivant=courant->suivant->suivant;

            else courant->suivant=NULL;

        }

        else {

            L=NULL;

        }

    }

}

```

// Qution 2

```

void filtre(struct Liste* L){

    struct Liste *courant;

    struct Liste*courantcourant=L;

    while(courantcourant!=NULL){

        courant=courantcourant->suivant; // pour supprimer le deuxième élément

        if(courant !=NULL){

            courantcourant->suivant=courant->suivant;

            free(courant);

        }

    }

}

```

```
}
```

```
courantcourant=courantcourant->suivant;
```

```
}
```

```
}
```