
COMPTE RENDU TP 5 RN40

BARTHELME ALEXANDRE

Mon code complet est disponible sur mon GitHub à l'adresse :
<https://github.com/Iron68A/RN40/tree/main/TP/TP5>

Ou par capture d'écran

Exercice 1 :

```
int u(int n){
    if(n==0){
        return 1;
    }
    else{
        return 2*u(n-1)+3*v(n-1)+w(n-1);
    }
}

int v(int n){
    if(n==0){
        return 2;
    }
    else{
        return u(n-1)+v(n-1)+2*w(n-1);
    }
}

int w(int n){
    if(n==0){
        return 3;
    }
    else{
        return u(n-1)+4*v(n-1)+w(n-1);
    }
}

int main(){
    int n = 1;
    int r= u(n);
    printf("%d",r);
    return 0;
}
```

Exercice 2 :

```
void inverse(void)
{
    char c;
    if((c = getchar()) != '.'){ inverse(); }
    putchar(c);
    return;
}
```

Exercice 3 :

Compter les éléments d'une liste :

```
20 int compter (liste l)
21 {
22     if (est_vide(l))
23         return 0;
24     else
25         return 1 + compter(reste(l));
26 }
```

Tester si une liste est triée par ordre croissant :

```
✓ int est_triee(liste l){
✓     if(l==NULL){
        return TRUE;
    }
    ✓ else if(l->suivant==NULL){
        return TRUE;
    }
    ✓ else if(l->val>l->suivant->val){
        return FALSE;
    }
    ✓ else{
        return est_triee(l->suivant);
    }
}
```

Concaténer deux listes (sans prendre en compte un trie) :

```

87 //concatener deux listes de manière récursive
88 liste concatener(liste l1, liste l2){
89     liste l3;
90     if(l1==NULL){
91         return l2;
92     }
93     else if(l2==NULL){
94         return l1;
95     }
96     else{
97         l3=l1;
98         l3->suivant=concatener(l1->suivant,l2);
99         return l3;
100     }
101 }
102 }

```