

# Projet

## Algorithmique et Structure de données

20\_12\_2020

---

ATALLA Salim  
NISO Aram

L2 info – groupe 387O

## Question 0)

### // Jeux de tests pour Q0:

```
cout << "----- Q0 -----" << endl;
t_maillon *Q0 = new t_maillon;
Q0->suivant = nullptr;

cout << "----- integration -----" << endl;
ajouterDebut("2", Q0);
ajouterDebut("1", Q0);
ajouterFin("3", Q0);
ajouterFin("4", Q0);
afficherM(Q0);

cout << "----- suppression -----" << endl;
supprimerDebut(Q0);
supprimerFin(Q0);
afficherM(Q0);
```

```
----- Q0 -----
----- integration -----
1, 2, 3, 4
----- suppression -----
2, 3
```

Cette première question était plutôt une question classique pour gérer une liste des chaînes de caractères et donc la complexité de la question est basique donc une fois que l'on sait gérer les listes chaînées alors la question devient normale.

## Question 1)

### // Jeux de tests pour Q1:

```
cout << "----- Q1 -----" << endl;
t_maillon* Q1;
Q1 = construireListe(5);
afficherM(Q1);

cout << "----- suppression -----" << endl;
suppressionPseudonyme(Q1);
afficherM(Q1);

cout << "----- recharger -----" << endl;
rechargerListe(Q1, 3, 5);
afficherM(Q1);
cout << "----- suppression début puis recharger -----" << endl;
```

// après la suppression d'un seul élément la procédure rechargerListe ne fonctionne pas car le nombre d'éléments reste supérieur du seuil.

```
supprimerDebut(Q1);
rechargerListe(Q1, 3, 5);
afficherM(Q1);
```

D'après les jeux de tests alors notre code semble fonctionnel et en plus notre code nous renvoie les résultats attendues.

```
----- Q1 -----
xeqy76, byju55, seco75, nabo20, nage55
----- suppression -----

----- recharger -----
guga78, nycy62, vyme44, peko03, gyqi60
----- suppression début puis recharger -----
nycy62, vyme44, peko03, gyqi60
```

## Question 2)

### // Jeux de tests pour Q2:

```
cout << "----- Q2 -----" << endl;
t_date date;
date.an = 2020;
date.mois = 10;
date.jour = 20;

string util = "Utilisateur";

t_contact contact1, contact2;
contact1.date = date;
contact1.pseudo1 = util; // l'utilisateur.
contact1.pseudo2 = donne_pseudo();

contact2.date = date;
contact2.pseudo1 = util; // l'utilisateur.
contact2.pseudo2 = donne_pseudo();

t_maillon_contact *Q2 = new t_maillon_contact;
Q2->suivant = nullptr;

cout << "----- Integration -----" << endl;
integrationContact(Q2, contact1);
integrationContact(Q2, contact2);
afficherMC(Q2, date);

t_date Aujourd'hui;
Aujourd'hui.an = 2020;
Aujourd'hui.mois = 11;
Aujourd'hui.jour = 25;

cout << "----- suppression -----" << endl;
SuppPlus2Sem(Q2, Aujourd'hui);
afficherMC(Q2, Aujourd'hui);
```

D'après les jeux de tests alors  
notre code semble fonctionnel  
et en plus notre code nous  
renvoie les résultats attendues.

```
----- Q2 -----
----- Integration -----
A la date de 20/10/2020, Utilisateur croise jofe14 et geky31
----- suppression -----
La liste est vide!
```

## Question 3) a)

```
struct t_utilisateur{

    string nom; // le nom de l'utilisateur, que l'on va prendre dans la liste de
                // contacts. Et nous avons choisi cette variable pour que
                // l'utilisateur soit reconnaissable dans la liste.

    bool covid; // si l'utilisateur a le covid ou non. (donc sous la forme d'un
                // booléen), et nous avons choisi cette variable enfin pour
                // différencier les utilisateurs qui ont le covid de ceux qui
                // l'ont pas.

};
```

```
//-----
struct t_tous_contacts{ // liste de taille 14 max.

    t_utilisateur utilisateur;
    t_maillon_contact *liste_contact; // liste de taille 20 max.
    t_tous_contacts *suivant;
};
-----
```

b + c)

// Jeux de tests pour Q3:

```
cout << "----- Q3 -----" << endl;
// créer des dates.
t_date date1;
date1.an = 2020;
date1.mois = 10;
date1.jour = 20;

t_date date2;
date2.an = 2020;
date2.mois = 11;
date2.jour = 12;

t_date Aujourd'hui;
Aujourd'hui.an = 2020;
Aujourd'hui.mois = 11;
Aujourd'hui.jour = 20;

// Créer un utilisateur.
t_utilisateur util;
util.nom = "utilisateur";
util.covid = false;

// créer des listes de contactes.
t_maillon_contact *MC1 = new t_maillon_contact;
MC1->suivant = nullptr;
t_maillon_contact *MC2 = new t_maillon_contact;
MC2->suivant = nullptr;

// créer des contactes, et les intégrer dans leurs listes.
t_contact contact1, contact2;
contact1.date = date1;
contact1.pseudo1 = util.nom; // l'utilisateur.
contact1.pseudo2 = donne_pseudo();

contact2.date = date1;
contact2.pseudo1 = util.nom; // l'utilisateur.
contact2.pseudo2 = donne_pseudo();

integrationContact(MC1, contact1);
integrationContact(MC1, contact2);

contact1.date = date2;
contact1.pseudo1 = util.nom; // l'utilisateur.
contact1.pseudo2 = donne_pseudo();

contact2.date = date2;
contact2.pseudo1 = util.nom; // l'utilisateur.
contact2.pseudo2 = donne_pseudo();

integrationContact(MC2, contact1);
integrationContact(MC2, contact2);
```

```
// Créer une liste de tous les contacts.
t_tous_contacts *Q3 = new t_tous_contacts;
Q3->suivant = nullptr;

cout << "----- integration -----" << endl;
IntegrationListeContact(Q3, util, MC1);
IntegrationListeContact(Q3, util, MC2);
afficherLTC(Q3);

cout << "----- suppression plus 2 semaines -----" << endl;
SuppListePlus2Sem(Q3, Aujourd'hui);
afficherLTC(Q3);
```

D'après les jeux de tests alors  
notre code semble fonctionnel et  
en plus notre code nous renvoie  
les résultats attendues.

```
----- Q3 -----
----- integration -----
A la date de 12/11/2020, utilisateur croise mifa05 et lyby52
A la date de 20/10/2020, utilisateur croise hydy07 et kuva48
----- suppression plus 2 semaines -----
A la date de 12/11/2020, utilisateur croise mifa05 et lyby52
```

#### Question 4)

Pour cette dernière question, nous avons réussi à refaire le scénario ( voir à l'intérieur du code ), nous avons pu mettre un message alertant que tel personne a le COVID enfin pour faire 'attention'.

Notre code semble fonctionnel mais pas satisfaisant parce que nous savons bien mettre une alerte 'un message' mais nous ne savons pas comment le mettre en lien avec tous les autres contacts.

Nous avons pensé à créer une variable 'Alerte' de type bool et donc cette variable sera 'true' une fois que un cas de covid est détecté et donc elle envoie une alerte aux contacts de cette personnes mais nous avons pas su comment la mettre en lien avec la liste de contacts.

```
A la date de 20/12/2020, Alice croise Charles
A la date de 20/12/2020, Charles croise Alice
ATTENTION: Alice a le covid!
```

#### Conclusion:

On conclut en disant que l'on a fait ce que l'on savait, notre code semble fonctionnel, il nous donne les résultats attendus mais peut être pas complet.

Notre plus grande difficulté était le temps, parce qu'avec tout ce que l'on avait à faire alors ce n'est pas facile de s'organiser mais enfin on a pu s'organiser et finir nos projets.

Concernant le code même, alors nous avons vu un peu de difficulté à utiliser les pointeurs mais grâce au cours et aux questions posées aux enseignants alors nous avons pu comprendre cela et donc compliqué notre projet.