

## **PROBLEM (COVID-19)**

A vaccination program has been launched to vaccinate a large group of citizens in order to guarantee herd immunity against the new coronavirus (COVID-19), which requires effective management of the vaccination process and monitoring of vaccinated citizens. The following types of vaccine have been approved: AstraZeneca and Sinopharm. Immunity against the new coronavirus requires two doses of the vaccine used, with the second dose administered two to four weeks after the first.

Vaccinated citizens are managed in a **table of structures**, each structure representing a citizen. Citizen information includes National identification number (cin), surname (nom), first name (prenom), vaccine type (type), date of first dose (date\_prem), and number of days between first and second dose (nbj).

The following structures are taken into account:

```

typedef struct {
int jour ;
int mois ;
int annee ;
} Date ;

typedef struct {
char type[30] ;
Date date_prem ;
int nbj ;
} Vaccin ;

typedef struct {
char cin[30] ;
char nom [30] ;
char prenom[30];
Vaccin vac ;
} Citoyen ;

#define max 100

```

=>The functions to be written are as follows:

- 1) void lecture (Citoyen C[max] , int N) which is used to enter information about the citizens concerned by the vaccination. N represents the total number of citizens.
- 2) void affichage (Citoyen C[max] , int N) which displays information for concerned by the vaccination. N represents the total number of citizens.
- 3) void totalVaccin (Citoyen C[max] , int N) which calculates and displays the number of citizens who will take a type of vaccine, where C is a table containing the citizens concerned by the vaccination. N represents the total number of citizens.
- 4) void CitoyenVac (Citoyen C[max] , int N, char type[30],Citoyen tab[max], int \* k) which fills a citizens table ( tab ) containing the citizens who will take the vaccine specified

by the type (type). C represents an array of citizens, N represents the total number of citizens and k is used to store the size of the array (tab).

5) `Date * deuxDose (Citoyen C[max] , int N, char type[30])` which returns an array of type `Date` (a pointer to `Date`) containing the dates of the second dose for the citizens concerned by a type of vaccine specified by the function's third argument. C represents an array of citizens and N represents the total number of citizens.

6) `void AfficheDate(Citoyen C[max] , int N, char cin[20] )` which displays the dates of the two scheduled doses for a citizen identified by CIN. C represents an array of citizens and N represents the total number of citizens.

## Correction

1)

```
void Lecture(Citoyen C[max],int N){
    int i;
    for ( i=0; i<N ; i++)
    { printf("Entrer Le CIN :"); scanf("%s",C[i].cin);
      printf("Entrer Le nom :"); scanf("%s",C[i].nom);
      printf("Entrer Le prenom :"); scanf("%s",C[i].prenom);
      printf("Entrer le type de vaccin :"); scanf("%s",C[i].vac.type);
      printf("Entrer la date de vaccin \n");
      printf(" Entrer le jour :"); scanf("%d",&C[i].vac.date_prem.jour);
      printf(" Entrer le mois :"); scanf("%d",&C[i].vac.date_prem.mois);
      printf(" Entrer l'annee :"); scanf("%d",&C[i].vac.date_prem.annee);
      printf("Entrer Le nombre de jours entre les deux doses :");
      scanf("%d",&C[i].vac.nbj);
      printf("*****\n");
    }
}
```

2)

```
void Affichage(Citoyen C[max],int N){
    int i ;
```

```

for ( i=0; i<N ; i++)
{ printf("Le CIN : %s\n",C[i].cin);
  printf("Le nom : %s\n",C[i].nom);
  printf("Le prenom :%s\n",C[i].prenom);
  printf("Le type de vaccin : %s\n",C[i].vac.type);
  printf("La date de vaccin: %d/%d/%d\n ", C[i].vac.date_prem.jour ,
C[i].vac.date_prem.mois, C[i].vac.date_prem.annee);
  printf("Le nombre de jours entre les deux doses : %d\n",C[i].vac.nbj);
  printf("*****\n");
}
}

```

3)

```

void totalVaccin (Citoyen C[max] , int N)
{ int i,Nb_Astra=0,Nb_Sino=0 ;
  for ( i=0; i<N ; i++)
  { if(strcmp(C[i].vac.type,"Astrazeneca")==0)
    Nb_Astra++;
    else Nb_Sino++;
  }
  printf("Le nombre de citoyens qui vont Astrazenica %d\n",Nb_Astra);
  printf("Le nombre de citoyens qui vont Sinopharme %d\n",Nb_Sino);
}

```

4)

```

void CitoyenVac (Citoyen C[max],int N,char type[30],Citoyen tab[max],int * k)
{ Citoyen tab[max];
  int i;
  *k=0;
  for ( i=0; i<N ; i++)
  { if(strcmp(C[i].vac.type,type)==0)
    { tab[*k] = C[i];
      (*k)++;
    }
  }
}

```

5)

```

Date * deuxDose(Citoyen C[max],int N ,char type[30])
{ int i,j=0;

```

```

Date dates[max];
for ( i=0; i<N ; i++)
{ if(strcmp(C[i].vac.type,type)==0)
  { dates[i].jour = ( C[i].vac.date_prem.jour + C[i].vac.nbj )%30;
    dates[i].mois = ( C[i].vac.date_prem.mois +(C[i].vac.date_prem.jour +
C[i].vac.nbj)/30)%12;
    dates[i].annee = C[i].vac.date_prem.annee +(C[i].vac.date_prem.mois
+(C[i].vac.date_prem.jour + C[i].vac.nbj)/30)/12;
  }
}
return dates;
}

```

6)

```

void AfficheDate(Citoyen C[max],int N,char cin[20])
{ int i,j=0;
  Date d;
  for ( i=0; i<N ; i++)
  { if( strcmp(C[i].cin,cin)==0)
    {
      d.jour =(C[i].vac.date_prem.jour + C[i].vac.nbj )%30;
      d.mois = ( C[i].vac.date_prem.mois +(C[i].vac.date_prem.jour + C[i].vac.nbj)/30)%12;
      d.annee = C[i].vac.date_prem.annee +(C[i].vac.date_prem.mois
+(C[i].vac.date_prem.jour + C[i].vac.nbj)/30)/12;
      printf(" %s %s %s \n",C[i].nom,C[i].prenom,cin);
      printf("La date de la premiere dose:
%d/%d/%d\n",C[i].vac.date_prem.jour,C[i].vac.date_prem.mois,C[i].vac.date_prem.annee
);
      printf("La date de la deuxieme dose: %d/%d/%d",d.jour,d.mois,d.annee);
    }
  }
}

```