

# Donne moi ton numéro de carte bancaire

Ce sujet de projet vous demandera l'application des connaissances vues en cours, travaux dirigés et travaux pratiques d'architecture des ordinateurs.

## 1 Modalités

### 1.1 Directives générales

1. Ce projet est à faire en binôme, impérativement.
2. Le programme doit être écrit en assembleur MIPS grâce au simulateur Mars 4.5 (fourni en TP).
3. Le programme doit être commenté et édité selon les bonnes pratiques vues en TP (et disponibles sur Moodle)
4. Le travail à remettre comprend, pour chaque groupe :
  - Un unique fichier `codeMIPS_GROUPETP_NOM1_NOM2.s` où
    - `GROUPETP` est le numéro de groupe de TP du binôme, et
    - `NOM1`, et `NOM2` le nom de chaque membre du groupe (dans l'ordre alphabétique)
  - Un rapport `rapportMIPS_GROUPETP_NOM1_NOM2.pdf` de deux pages maximum qui détaille :
    - Vos noms, prénom et groupe de TP (pas de page de garde).
    - Les choses qui fonctionnent dans votre programme, le cas échéant ;
    - Les choses qui ne fonctionnent pas dans votre programme, le cas échéant ;
    - La manière dont vous vous êtes organisés pour travailler en binôme ;
    - Tout complément d'information que vous jugeriez utile (point intéressant, amélioration, originalité, etc.), pourvu que la limite de deux pages ne soit pas dépassée.

### 1.2 Dates clefs

Événement	Date limite	Support
Choix du groupe	24 février 2020	Mail
Remise des travaux	11 mars 2020	Moodle

Avant la date limite relative au choix du groupe, vous enverrez un mail à [scholet@unistra.fr](mailto:scholet@unistra.fr) afin de préciser

- votre groupe de TP
- les nom et prénom de chacun des membres du groupe

Les éléments à remettre (identifiés au 1.1.4) doivent être archivés dans un fichier \*.zip, qui doit ensuite être déposé sur Moodle, dans l'espace prévu à cet effet, par l'un des membres du groupe (pas les deux), avant la date limite correspondante.

## 2 Travail à faire

### 2.1 Fonctionnalités du programme

Le programme devra avoir au mieux les trois fonctionnalités suivantes :

1. Valider un numéro de carte bancaire, i.e. dire si le numéro est valide (choix "1");
2. Générer un numéro de carte bancaire valide (choix "2");
3. Quitter (choix "3").

Au lancement, l'utilisateur est invité à faire le choix d'une fonctionnalité.

S'il fait le choix "1", il est invité à saisir un numéro de carte bancaire. Le programme affiche ensuite si le numéro est valide ou non.

S'il fait le choix "2", le programme lui affiche un numéro de carte bancaire valide.

S'il fait le choix "3", le programme quitte.

A l'issue du résultat affiché aux choix "1" ou "2", le programme ré-invite l'utilisateur à faire le choix d'une fonctionnalité (par conséquent, le programme ne quitte pas).

## 2.2 Contraintes

Le programme devra impérativement implémenter l'algorithme du modulo 10 (voir plus bas).

Pour la validation d'un numéro de carte bancaire, le programme ne devra pas avoir des numéros valides "pré-enregistrés".

Pour la génération d'un numéro de carte bancaire, le programme ne devra pas utiliser de numéros "pré-enregistrés" : il devra générer ce numéro via des calculs.

Le programme ne devra quitter que si l'utilisateur fait le choix "3".

Le programme ne devra pas utiliser de macros (`.macro ... .end_macro`).

## 2.3 Améliorations possibles

Ces propositions constituent une liste non-exhaustive d'améliorations que vous pouvez implémenter. Toute tentative suffisamment aboutie aura un impact positif au regard de votre note. Vous avez également la possibilité d'implémenter une amélioration non listée ci-dessous, à l'exception de toute amélioration relative aux codes CVV/CID/4DBC (numéros inscrits au dos de la carte). Dans tous les cas, si vous implémentez une amélioration, vous devrez l'identifier clairement dans votre programme (via des commentaires) et dans votre rapport.

- Lors de la validation d'un numéro de carte bancaire (choix "1"), si le numéro est valide, afficher également le type de carte associé au numéro (Visa, MasterCard, Diners Club, ...).
- Lors de la génération d'un numéro valide de carte bancaire (choix "2"), laisser le choix à l'utilisateur du type de carte dont le numéro doit être généré.

Note : pour les améliorations liées au type de carte, il est autorisé d'enregistrer en mémoire les premiers chiffres liés à l'émetteur (et seulement les premiers chiffres liés à l'émetteur) d'un numéro valide.

# 3 Ressources et recommandations

## 3.1 Ressources

### 3.1.1 Validité d'un numéro

Un numéro valide de carte bancaire est un numéro qui respecte l'algorithme "du modulo 10" (Voir sur Wikipédia). Par exemple, dans un numéro de 15 chiffres, le 15<sup>e</sup> chiffre sert à valider les autres, et est calculé comme suit :

1. Omettre le dernier chiffre (c'est celui qu'on cherche à vérifier)
2. Inverser l'ordre des chiffres
3. Multiplier les chiffres dont la position est impaire par 2
4. Soustraire 9 aux nombres strictement supérieurs à 9
5. Sommer les chiffres
6. Calculer le modulo 10 de la somme
7. Le dernier chiffre est le résultat précédent moins 10

Voici un exemple avec le numéro American Express 376701449043032 (15 chiffres) :

Etape	Chiffres														Dernier chiffre
Numéro	3	7	6	7	0	1	4	4	9	0	4	3	0	3	2
1.	3	7	6	7	0	1	4	4	9	0	4	3	0	3	
2.	3	0	3	4	0	9	4	4	1	0	7	6	7	3	
3.	6	0	6	4	0	9	8	4	2	0	14	6	14	3	
4.	6	0	6	4	0	9	8	4	2	0	5	6	5	3	
5.	Somme = 58														
6.	58 Modulo 10 = 8														
7.	10 - 8 = 2														2

Si le 15<sup>e</sup> chiffre calculé via l'algorithme est celui du numéro testé, alors ce numéro est valide. Le cas contraire, il ne l'est pas. Dans le cas plus haut, le dernier numéro (le 15<sup>e</sup>) calculé est bien celui testé (i.e. 2), alors le numéro American Express 376701449043032 est valide.

A partir de cet algorithme, il est plutôt facile de vérifier la validité d'un numéro. Du coup, comment générer un numéro valide ?

### 3.1.2 Emetteurs de carte bancaire

Les cartes bancaires sont aussi référencées selon leur émetteur (Visa, MasterCard, Diners Club, ...), qu'on appellera "type" dans ce projet. Les émetteurs se partagent les numéros grâce à l'attribution de plages de numéros, un peu comme les pays se partagent des plages d'adresses IP.

Voici les principales correspondances entre émetteur et plage de numéros :

Emetteur	Numéro débute par ou est dans la plage	Nombre de chiffres
American Express	34, 37	15
Diners Club - International	36	14
Discover	6011, 644, 645, 646, 647, 648, 649, 65, 622126 à 622925	16 à 19
InstaPayment	637, 638, 639	16
JCB	3528 à 3589	16 à 19
Maestro	5018, 5020, 5038, 2893, 6304, 6759, 6781, 6762, 6763	16 à 19
MasterCard	51, 52, 53, 54, 55 222100 à 272099	16
Visa	4	13, 16 ou 19
Visa Electron	4026, 417500, 4508, 4844, 4913, 4917	16

On note que dans bien des cas, le ou les premiers chiffres suffisent à déterminer le type de carte. Aussi, peu importe le nombre de chiffres d'un numéro, le processus de validation reste inchangé.

### 3.1.3 Tests du programme

Réalisez des tests réguliers de votre programme : n'attendez pas la fin de sa réalisation. En particulier, pour tester si un numéro est valide ou non, déterminez un certain nombre de numéros valides à l'avance, puis utilisez-les pour vos tests. Quelques exemples de numéros valides :

Emetteur	Numéro
American Express	376701449043032, 346289151163344
Diners Club - International	36115979031808, 36183207442825
Discover	6011769148196011, 6011374768589116236
InstaPayment	6373370803365249, 6392412456390228
JCB	6385318570064111, 3545157477293116903
Maestro	5018076893191931, 6759446751016119
MasterCard	2720997803956293, 5263240318336578
Visa	4629829104859538, 4468228737634471
Visa Electron	4913732226176374, 4917293399840631

## 3.2 Recommandations

Commencez par bien prendre en main le sujet en l'étudiant de manière approfondie : soyez sûr d'avoir compris ce qui est demandé. Ne vous lancez pas tout de suite dans le code assembleur : concevez votre solution à l'aide de schémas et d'algorithmes pour y voir plus clair. Cela vous aidera aussi à lister les routines (fonctions) que vous devez faire, et vous aidera à mieux partager le travail.

Documentez-vous sur les numéros de carte bancaire, le net regorge de ressources à ce sujet. Faites tourner l'algorithme du modulo 10 à la main (ou sur Excel/Calc) pour être sûr de son fonctionnement : c'est le pilier de votre programme !

Ce travail doit faire l'objet de travaux en dehors des heures de TP. Néanmoins, votre intervenant de TP reste votre interlocuteur privilégié pour toute aide ou question relative au TP (par mail de préférence, ou en marge des TP). Ne restez pas bloqués !

Placez des commentaires pertinents : il est inutile de commenter chacune des lignes ! En revanche, dire ce que fait une routine juste au dessus peut s'avérer utile.

La notation prendra en compte, de manière non-exhaustive :

- La lisibilité du code produit et le respect des bonnes pratiques d'édition (commentaires, indentation, etc.)
- L'implémentation de l'algorithme du modulo 10
- Le nombre de choix implémentés
- La capacité du binôme à travailler ensemble
- Le respect des éléments requis dans le rapport
- Les éventuelles améliorations apportées au programme

```
.data
    msg:
        .ascii "Bon travail ! \n"

.text

    la $a0 msg
    li $v0 4
    syscall

    j exit

exit :
    li $v0 10
    syscall
```