
TP1 : ROMANISATION DE TEXTE EN CORÉEN.

Introduction

Pour le premier tp vous allez construire un logiciel qui traduit du coréen en écriture latine (romanisation ou latinisation). L'utilisateur va entrer le coréen au clavier sous forme d'Unicode. Le logiciel va ensuite traduire le texte en caractères latins. Le résultat sera affiché à l'écran.

Description

Entrées

L'utilisateur va écrire les phrases à traduire au clavier. Chaque caractère coréen (Hangeul) est entré sous forme d'Unicode prêt à être affiché dans une page web. Chaque valeur sera précédée des caractères esperluette ('&') et ('#') et sera suivi du caractère (';'). Le texte peut contenir des points '.', dans ce cas ils sont situés entre deux caractères. Il peut y avoir des espaces entre chaque caractère et/ou les '.'. Les entrées se terminent par la fin de ligne.

Par exemple :

퉔 띠 쁘 클 띠 끈 유 울 똔 을
쁘 몰 파 였 고 .

S'il y a une erreur lors de l'entrée des données alors vous affichez un message d'erreur et le logiciel termine.

Analyse

Le logiciel va traduire cette suite en alphabet latin. Cela va se faire en trois (3) étapes : extraire les jamos, traduire en alphabet latin et finalement faire quelques correctifs sur la prononciation dépendant de la position des jamos dans le texte.

Hangeul et Jamo : les jamos sont les 'lettres' coréennes. Ils représentent les sons de base. Un Hangeul (caractère coréen) est composé de 2 ou 3 jamos. Comme dans plusieurs alphabets, les jamos sont divisés en deux catégories : les consonnes et les voyelles.

Extraction des jamos

Le premier jamos d'un hangeul est une consonne (C_i), il est suivi d'une voyelle (v) et finalement il peut y avoir une consonne finale (C_f). Pour trouver ces trois jamos il suffit d'inverser la formule suivante à partir de l'Unicode (U_C).

$$U_C = 44032 + 588 \times C_i + 28 \times v + C_f$$

Si la valeur pour la consonne finale vous donne 0 c'est qu'il n'y a pas de consonne finale.

Romanisation

Pour cette étape il suffit de traduire chaque jamos en alphabet latin. Le tableau suivant contient la traduction de la première consonne :

Nombre (C_i)	Romanisation	Ja-eum
0	g	ㄱ
1	kk	ㄲ
2	n	ㄴ
3	d	ㄷ
4	tt	ㄸ
5	r	ㄹ
6	m	ㅁ
7	b	ㅂ
8	pp	ㅃ
9	s	ㅅ
10	ss	ㅆ
11	(rien)	ㅇ
12	j	ㅈ
13	jj	ㅉ
14	ch	ㅊ
15	k	ㅋ
16	t	ㅌ
17	p	ㅍ
18	h	ㅎ

Voici la traduction pour les voyelles :

Nombre (v)	Romanisation	Mo-eum
0	a	ㅏ
1	ae	ㅑ
2	ya	ㅓ
3	yae	ㅕ
4	eo	ㅗ
5	e	ㅛ
6	yeo	ㅜ
7	ye	ㅠ
8	o	ㅡ
9	wa	ㅜ
10	wae	ㅟ
11	oe	ㅝ
12	yo	ㅠ
13	u	ㅜ
14	wo	ㅟ
15	we	ㅞ
16	wi	ㅟ
17	yu	ㅠ
18	eu	ㅡ
19	ui	ㅟ
20	i	ㅣ

Finalement, voici le tableau pour la dernière consonne. Pour les trois tableaux, la première colonne indique le nombre utilisé pour calculer l'UNICODE, la deuxième colonne contient la romanisation que votre programme doit retourner et la dernière colonne contient le caractère coréen.

Nombre (C_f)	Romanisation	Ja-eum
0	(rien)	Pas de lettre
1	k	ㄱ
2	k	ㄲ
3	kt	ㅋ
4	n	ㄴ
5	nt	ㄷ
6	nh	ㄸ
7	t	ㄷ
8	l	ㄹ
9	lk	ㄺ
10	lm	ㄻ
11	lp	ㄼ
12	lt	ㄽ
13	lt	ㄾ
14	lp	ㄿ
15	lh	ㅀ
16	m	ㅁ
17	p	ㅂ
18	pt	ㅃ
19	t	ㅅ
20	t	ㅆ
21	ng	ㅇ
22	t	ㅈ
23	t	ㅊ
24	k	ㅋ
25	t	ㅌ
26	p	ㅍ
27	h	ㅎ

Ajustement

Ensuite, il y a quelques ajustements sur la prononciation. La romanisation d'une consonne finale (C_i) est modifiée selon la consonne initiale (C_{i+1}) du prochain Hangeul. Le tableau suivant contient les modifications possibles. Les cases écrites en rouge contiennent les cas où il y a une modification. Les cas non présents dans la table n'ont pas de modification.

$C_i \backslash C_{i+1}$		ㅇ	ㄱ	ㄴ	ㄷ	ㄹ	ㅁ	ㅂ	ㅅ	ㅈ	ㅊ	ㅋ	ㅌ	ㅍ	ㅎ
		(rien)	g	n	d	r	m	b	s	j	ch	k	t	p	h
ㄱ	k	g	kg	ngn	kd	ngn	ngm	kb	ks	kj	kch	k-k	kt	kp	kh
ㄴ	n	n	n-g	nn	nd	ll	nm	nb	ns	nj	nch	nk	nt	np	nh
ㄷ	t	d	tg	nn	td	nn	nm	tb	ts	tj	tch	tk	t-t	tp	th
ㄹ	l	r	lg	ll	ld	ll	lm	lb	ls	lj	lch	lk	lt	lp	lh
ㅁ	m	m	mg	mn	md	mn	mm	mb	ms	mj	mch	mk	mt	mp	mh
ㅂ	p	b	pg	mn	pd	mn	mm	pb	ps	pj	pch	pk	pt	p-p	ph
ㅅ	s	sg	ngn	td	nn	nm	tb	ts	tj	tch	tk	t-t	tp	th	
ㅇ	ng	ng-	ngg	ngn	ngd	ngn	ngm	ngb	ngs	ngj	ngch	ngk	ngt	ngp	ngh
ㅈ	t	j	tg	nn	td	nn	nm	tb	ts	tj	tch	tk	t-t	tp	th
ㅊ	t	ch	tg	nn	td	nn	nm	tb	ts	tj	tch	tk	t-t	tp	th
ㅋ	t	t	tg	nn	td	nn	nm	tb	ts	tj	tch	tk	t-t	tp	th
ㅌ	h	h	k	nn	t	nn	nm	p	hs	ch	tch	tk	tt	tp	t

Sorties

Le programme termine en affichant la suite des caractères. Les points sont aussi affichés au même endroit qu'il était dans le texte. Seul le résultat est affiché, aucun autre texte ne doit être affiché.

Exemple complet

Soit les entrées suivantes :

갇 퉔 띠 쁘 클 띠 끈 유 울 똔
을 쁘 몰 파 였 고 .

Décomposition en jamos :

(0, 0, 7), (16,14,12), (4,20,0), (8,18,0), (15,18,8), (4,20,0), (1,18,4), (11,17,0), (11,13,8), (4,8,4), (11,18,8),
(8,18,0), (6,8,8), (17,0,0), (11,6,20), (0,8,0).

Romanisation :

(g, a, t), (t, wo, lt), (tt, i), (pp, eu), (k, eu, l), (tt, i), (kk, eu, n), (yu), (u, l), (tt, o, n), (eu, l), (pp, eu), (m, o, l),
(p, a), (yeo, t), (g, o).

Ajustement :

(g, a, t-), (t, wo, lt), (tt, i), (pp, eu), (k, eu, l), (tt, i), (kk, eu, n), (yu), (u, l), (tt, o, n), (eu, l), (pp, eu), (m, o, l),
(p, a), (yeo, t), (g, o).

Affichage :

gat-twoltttippeukeulttikkeunyuulttoneulppeumolpayeotgo.

Construction

Chacune de vos fonctions doit avoir des commentaires. Ces commentaires doivent contenir l'information suivante :

- Rôle de la fonction : ce qu'elle fait, ce qu'elle retourne.
- Argument : Description des arguments, leurs rôles dans la routine, les valeurs permises, sont-ils des entrées ou des sorties, ...
- Les erreurs possibles.

Remise

Remettez votre programme en utilisant Moodle. Votre programme doit être dans un seul fichier.