

Université de Nice-Sophia Antipolis Licence Informatique - 2e année U.E. Introduction au Langage C 2017-2018

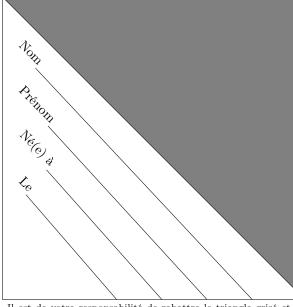
> Épreuve d'Examen Terminal du Vendredi 18 Mai 2018

Durée: 1 heure 30

Tous documents autorisés. Il est interdit d'accéder à internet ou d'utiliser un compilateur C



Toutes les questions sont indépendantes. Tous les codes sources devront être écrits en langage C.



Il est de votre responsabilité de rabattre le triangle grisé et de le cacheter au moyen de colle, agrafes ou papier adhésif. Si ne vous le faites pas, vous acceptez implicitement que votre copie ne soit pas anonyme.

1 HashCode 2018 – Self-driving rides (10 points)

Des millions de personnes se rendent chaque jour en voiture, à l'école ou sur leur lieu de travail. Les véhicules autonomes sont un développement passionnant pour le transport. Dans ce problème, nous verrons comment une flotte de véhicules autonomes peut efficacement transporter les passagers vers leurs destinations dans une ville virtuelle.

1.1 Quelques fonctions rapides

1. On vous donne la structure Point permettant de représenter un point par ses coordonnées cartésiennes.

```
struct Point {
    int x;
    int y;
};
Écrivez la fonction int manhattan (struct Point *p1, struct Point *p2), qui calcule et renvoie la distance de Manhattan entre les points p1 et p2. La distance de Manhattan entre (x, y) et (a, b) est égale à |x - a| + |y - b|. En C, la librairie stdlib.h fournit la fonction int abs (int x) qui renvoie la valeur absolue de x.
2. Supposons qu'il faille une unité de temps pour parcourir une unité de distance. Il faut 2 unités de temps pour aller de (0,0) à (1,1)(|0 - 1| + |0 - 1| = 2). Écrivez la fonction int startTime (struct Point *p1, int t, struct Point *p2) qui calcule et renvoie le temps auquel on arrive au point p2 en partant du point p1 au temps t.
```

	Quelques fonctions intermédiaires
1.	Une course est représentée par un numéro, un point de départ, un point d'arrivée, une heure de départ au plus tôt et une heure d'arrivée au plus tard. Le numéro, l'heure de départ et d'arrivée sont des entiers. Par exemple, $\{0, \{0, 0\}, \{1, 3\}, 2, 9\}$ correspond à la course 0 qui commence en $(0,0)$ et finit en $(1,3)$, elle commence au plus tôt au temps 2 et finit au plus tard au temps 9. Écrivez une structure Course permettant de représenter une course.
2.	Écrivez la fonction int waitTime (struct Point *p, int t, struct Course *c) qui calcule et renvoie le temps d'attente avant la course c en partant du point p au temps t. On suppose que c est la course 0 de la question précédente, en partant du point (0, 0) au temps 0, la fonction renvoie 2; en partant du point (1, 0) au temps 1, la fonction renvoie 0; en partant du point (2, 1) au temps 0, la fonction renvoie -1.
3.	Écrivez la fonction int rideTime (struct Point *p, int t, struct Course *c) qui calcule et renvoie le temps auquel on aura fini la course c en partant du point p au temps
	t. On suppose que c est la course 0 de la question 5, en partant du point $(0, 0)$ 0, la fonction renvoie 6; en partant du point $(1, 0)$ au temps 1, la fonction renv

	partant du point $(2, 1)$ au temps 0 , la fonction renvoie 7 .								
.3	Coeur du problème								
1.	On vous donne la structure suivante :								
	<pre>struct Vehicule {</pre>								
	struct Point cur;								
	<pre>int t_cur;</pre>								
	int nbc;								
	<pre>struct Course *courses; };</pre>								
	Un véhicule est donc représenté par sa position (cur) à un temps donné (t_cur), le nombre de courses (nbc) et le tableau des courses qu'il va effectuer (courses). Au départ, le véhicule se trouve à la position (0, 0) au temps 0 et a une liste de courses vide. Écrivez la fonction void initVehicule (struct Vehicule *v) qui initialise le véhicule passé en paramètre.								
	void initvenicule (struct venicule *v) qui initianse le venicule passe en parametre.								
2.	Écrivez la fonction int vehiculeRide (struct Vehicule *v, struct Course *c) qui calcule et renvoie le temps auquel on aura fini la course c avec le véhicule v ou 0 si on ne peut pas la finir avant l'heure d'arrivée au plus tard de c.								

1.3

3.	Écrivez la fonction struct Vehicule* assignRides (int nbv, int nbc, struct Course *courses) qui prend en paramètre le nombre de véhicules disponibles, le nombres de courses ainsi qu'un tableau de courses. Cette fonction crée et renvoie un tableau de véhi-
	cules, et essaye d'affecter chacune des courses à un véhicule.
	NB : Ne pas oublier de réallouer de la mémoire pour le tableau des courses des véhicules si
	nécessaire.
	-
	

2 T9 Dictionary (10 points)

Il y a fort fort longtemps avant l'ère des smartphones, les gens tapaient des messages sur les pavés numériques. Des lettres étaient assignées aux nombres en 2 (abc), 3 (def), 4 (ghi), 5 (jkl), 6 (mno), 7 (pqrs), 8 (tuv), 9 (wxyz). La façon la plus simple de produire du texte était de répéter en appuyant sur le bouton correspondant pour obtenir la lettre désirée.

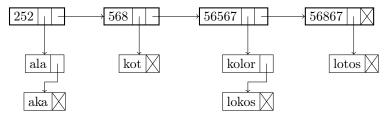
Cependant, même dans ces temps anciens, il y avait une méthode pour accélérer le processus entier. Il était possible de taper le mot comme des nombres et l'algorithme était capable de prédire le mot basé sur un dictionnaire. Par exemple le nombre 25 est lié à 9 mots : aj, ak, al, bj, bk, bl, cj, ck, cl. De même, le nombre 438 fait référence à 27 mots distincts, etc. Heureusement, l'existence du dictionnaire avec des "vrais mots" a rendu le nombre de possibilités raisonnablement bas.

2.1 Conversion

	•	ito") ren		

2.2 Dictionnaire

On utilise une liste chaînée pour le dictionnaire. La figure suivante montre un exemple de dictionnaire :



Cette liste est représentée par les structures suivantes :

```
typedef struct elm {
  char *mot;
  struct elm *suiv;
} Element;
typedef struct dico {
  unsigned int val;
  Element *liste;
  struct dico *next;
} Dico;
```

Element représente une liste de mot. Chaque composant contient un mot (mot) et un pointeur vers le suivant (suiv). Dico représente le dictionnaire. Chaque composant contient une valeur (val), une liste (liste) des mots associé à cette valeur, et un pointeur vers le suivant (next).

d addWord ste chaînéd

Ecrivez une fon u dictionnaire	dico de faço	n à ce que	la liste soi	it triée dan	s l'ordre cr	oissant des	valeı