Examen final – TPGO / 2CS – ESI 2012/2013
Durée 1h30 – Doc. Interdits
(Justifier toutes vos réponses)

1) Expliquez pourquoi la programmation dynamique est-elle qualifiée de « méthode ascendante » ?

2) Quand est-ce que l'exploration d'un espace de recherche nécessite t-elle un parcours en largeur ?

3) En utilisant le théorème de Böhm et Jacopini, donnez une forme structurée du programme suivant :

```
etiq1 :  SI ( x > 10) Allerà etiq2 ;
         y ← y+1 ;
etiq4 :  x ← x+1 ;
         SI ( y <= 20) Allera etiq1 ;
         Allerà etiq3 ;
etiq2 :  x ← x-10 ;
         SI ( x<=10 ) Allerà etiq4 ;
         y ← y*2 ;
         Allerà etiq1 ;
etiq3 :  stop.
```

4) Donnez les plus faibles préconditions nécessaires pour que les énoncés suivants, soient des théorèmes dans le système formel de Hoare :

a- E { x ← y-x ; z ← y-1 ; y ← x+3 ; x ← z-y } (x = 3)
b- E { SI (x > y) z ← 2 ; SINON x ← x+1 ; SI (y = x) z ← z-1 FSI ; x ← y+1 FSI ; } (x=z)
c- E {x ← a ; y ← b ; TQ (y>1) x ← x*2 ; y ← y div 2 FTQ } (x = ab)

5) Comment montrer qu'un programme fonctionnel P calcule partiellement une fonction f en utilisant le théorème du point fixe ?

6) Montrez par la méthode du point fixe, que le programme fonctionnel ci-dessous calcule exactement le terme fib(n) de la suite de Fibonacci :

P(n) = [λn. SI (< n 2) 1 SINON (+ (P (- n 1)) (P (- n 2)))]

7) Donnez un programme purement logique définissant un prédicat 'pair(X)' évalué à vrai si X représente un entier naturel pair.
On représentera les entiers naturels, en programmation logique, par l'équation à point fixe suivante :

Entier = {zéro OU s(Entier)}
ainsi l'entier 3 sera représenté par le terme structuré : s(s(s(zéro)))

Rappel : Système Formel de Hoare

AFF : t(exp) {x ¬ exp} t(x)
IMP1: (E⇒F , F{P}S) |-- E{P}S
CND1: (E&B{P}S , E&¬B⇒S) |-- E {SI B: P FSI} S
CND2: (E&B{P}S , E&¬B{Q}S) |-- E { SI B : P SINON Q FSI } S
ITE: E&B{P}E |-- E {TQ B : P FTQ} E&¬B
IMP2: (E{P}F , F⇒S) |-- E{P}S
SEQ: (E{P}F , F{Q}S) |-- E{P;Q}S

Rappel : Suite de Fibonacci

fib(0) = 1 ; fib(1) = 1 ;
fib(n) = fib(n-1) + fib(n-2)   pour tout n >1