

Programme de Colle N°4 (Semaine du 09/10/2023)

7 octobre 2023

1 Questions de Cours

1. Soit une fonction qui prend en entrée deux entiers a et $n > 0$ et calcule a^n avec l'exponentiation naïve (contenant une boucle avec n itérations). Montrer que cette fonction termine.
2. Soit une fonction qui prend en entrée deux entiers a et $n > 0$ et calcule a^n avec l'exponentiation naïve (contenant une boucle avec n itérations). Montrer que cette fonction est correcte.
3. Donner la définition d'une fonction g majorée par f , minorée par f , de même ordre de grandeur que f et équivalente à f , avec f et g deux fonctions de \mathbb{N} dans \mathbb{N} .
4. Ecrire l'algorithme d'exponentiation rapide en utilisant la récursivité.
5. Soit `fib` une fonction qui prend en entrée n et renvoie u_n , où u_n est le n ème terme de la suite de Fibonacci ($u_0 = 0$, u_1 et $u_{n+2} = u_{n+1} + u_n$), avec une méthode récursive naïve (cf cours). Représenter l'arbre d'appel de `fib(5)`.

2 Cours

2.1 Chapitre 1

- Problèmes : spécification, instances (entrées) et sorties, préconditions. Problèmes de décision et d'optimisation.
- Algorithmes : spécification, effets de bords.
- Programmes et Langages de Programmation : langage de haut et bas-niveau, langage interprété ou compilé, langage impératif ou fonctionnel.

2.2 Chapitre 2

- Caractéristiques du Langage C. Structure d'un programme en C, fonction `main`.
- Variables et Types. Types élémentaires et opérations.
- Fonctions : Signature et Corps. Paramètres et Arguments. Portées des Variables, Passage en Valeur. Syntaxe des fonctions en C.
- Conditions. Syntaxe en C.
- Boucles non-bornées et boucles bornées. Syntaxe en C.

- Affichage et Saisie en C.
- Assertions en C.

2.3 Chapitre 3

- Terminaison d'un algorithme. Variant de boucle, preuves de terminaison.
- Correction partielle ou totale d'un algorithme. Invariant de boucle, preuve de correction.
- Complexité temporelle : définition. Nombre d'opérations élémentaires pour une entrée donnée. Taille d'une entrée, complexité dans le pire cas et en moyenne pour une entrée de taille n .
- Notations de Landau, complexité asymptotique : fonction majorée par/minorée par/de même ordre de grandeur que/équivalente à f .
- Complexités usuelles, règles de calcul sur les complexités. Sommes usuelles. Analyse de complexité.

2.4 Chapitre 4

- Définition d'une fonction récursive. Syntaxe en C. Définition récursive des fonctions factorielle et puissance (exponentiation version naïve et rapide)
- Cas terminal d'une fonction récursive. Arbre d'appels.
- Comparaison des méthodes récursives et itératives. Exemple d'application de la récursivité : tours de Hanoi.
- Syntaxe du langage OCaml : déclaration de variables, types. Conditions, déclarations locales. Déclaration de Fonctions, Filtrage par Motifs.