

Algorithmique et structures de données 2017-18

Contenu du cours 8 du 16.11.2017

1. Mise en commun l'apprentissage par problème sur le backtracking (retour sur trace)
2. Nombre de solutions pour le problème des N reines (connu jusqu'à 27 reines)
3. Le paquetage pile
Les exceptions PILE_VIDE, PILE_PLEINE
4. La gestion des exceptions
 - Traitement des erreurs indépendant du code lui-même
 - Exception prédéfinies :
CONSTRAINT_ERROR, STORAGE_ERROR, PROGRAM_ERROR
 - Le type exception ainsi que le paragraphe correspondant
 - Levée de l'exception : raise (pour signaler la détection d'une erreur)
 - Propagation de l'exception à travers la pile d'appel de sous-programmes
 - Traitement de l'exception en principe par le sous-programme responsable de l'erreur (i.e. ayant généré l'exception) pas celui qui la détecte (i.e. l'ayant levée)
 - Le bloc d'instructions : declare, begin, exception, end
5. Utilisation de piles

Permutation d'un ensemble de valeurs

```
type T_Tab is array(Integer range<>) of Integer;
procedure Permutation(Tab : in out T_Tab) is
    Temp : T_Tab(Tab'range);
    J : Integer := Temp'First;
    Tampon : T_Pile;
    Gen : Generator;
begin
    Reset(Gen);
    for I in Tab'Range loop
        if Random(Gen) < 0.5 then
            Empiler(Tampon, Tab(I));
        else
            Temp(J) := Tab(I);
            J := J+1;
        end if;
    end loop;
    while not Vide(Tampon) loop
        Depiler(Tampon, Temp(J));
        J := J+1;
    end loop;
    Tab := Temp;
end Permutation;
```

Le tri avec deux piles

6. Le tri rapide

Explication + exemple

Choix du pivot => dernier élément

Implémentation partielle du tri en Ada