

Bases de la programmation impérative

Ensimag 1A

1. Lancer de dés

On souhaite utiliser un ordinateur pour expérimenter à très grande échelle le lancer de dés à six faces.

Toutes les fonctions demandées sont à écrire récursivement.

- 1.1. Programmez une fonction `somme_aleatoire(nombre_des)` renvoyant la somme (des valeurs) du nombre de dés demandés (entier naturel), lancés aléatoirement (une seule fois).
- 1.2. On souhaite maintenant énumérer *tous* les lancers possibles pour un nombre de dés donné. Pour ce faire, on utilise un tableau `des` dont chaque case correspondra à un dé (et contiendra donc les valeurs de 1 à 6 au cours de l'énumération). On effectue une récurrence sur le *nombre* de dés *non encore lancés*. Programmez la fonction `lancers(des_restants, des)`, où `des` contient un lancer partiel et `des_restants` le nombre de dés restant à lancer, qui affiche sur la sortie standard toutes les combinaisons possibles. Écrire ensuite une fonction `tous_les_lancers(nombre_des)` qui appelle `lancers` avec des paramètres initiaux permettant d'énumérer toutes les combinaisons pour le nombre de dés spécifié.
- 1.3. Écrivez une fonction `lancers_somme_contrainte(des_restants, des, somme_demandee)` affichant *toutes* les combinaisons. Parmi elles, celles dont la somme vaut `somme_demandee` seront affichées entre `***`.
- 1.4. Écrivez une fonction `occurrences_somme(nombre_des, somme_demandee)` renvoyant le nombre de combinaisons du nombre de dés donné atteignant la somme demandée. On essaiera d'éviter d'énumérer toutes les combinaisons possibles.
- 1.5. On suppose disposer d'une fonction `fonction_validation` évaluant un ensemble de dés (prenant un tableau de dés en argument). Cette fonction classe les combinaisons de dés en combinaisons valides ou invalides selon un certain critère arbitraire ; elle renvoie `True` lorsqu'une combinaison doit être considérée comme valide et `False` dans le cas contraire. Proposez une fonction `combinaisons_valides(nombre_des, fonction_validation)` renvoyant pour le nombre de dés donné le nombre de combinaisons valides selon la fonction de validation donnée.
- 1.6. Reprendre les questions 1.2 à 1.5 en ne comptant que les lancers distincts (relativement à la permutation des dés).