Université de Strasbourg
Algorithmique et Programmation 2 Feuille de TP

Boucles et récursion...

1. **Devine un nombre**
   Écrire un programme C qui tire au sort un nombre de 1 à 100 et demande a l'utilisateur de le deviner. Si l'utilisateur a trouvé le nombre, alors le jeu s'arrête ; sinon l'ordinateur affiche 'plus petit' ou 'plus grand'. L'utilisateur a le droit à cinquante tentatives pour deviner le nombre.

   Pour cela utiliser les fonctions `rand`, `randr` de la bibliothèque standard et la fonction `hasard` comme dans l'exemple suivant :

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>

int hasard(int min, int max){
return (int) (min + ((float) rand() / RAND_MAX * (max - min + 1)));
}

int main(){
  srand(time(NULL));
  int alea;
  int min = 1;
  int max = 6;

  alea = hasard(min, max);

  printf("%d",alea);
  return 0;
}
```

2. **Jeu des allumettes**

   N allumettes sont positionnées initialement sur la table. Deux joueurs retirent entre 1 et p allumettes, chacun à leur tour. Celui qui retire la dernière allumette a gagné.

   a. Concevez et codez une fonction qui prend un entier R en paramètre, et qui affiche R barres verticales à l'écran.
   b. Concevez et codez une fonction qui permet de jouer : les joueurs choisiront le nombre d'allumettes par une entrée clavier.
   c. Modifiez votre fonction pour jouer seul contre un joueur virtuel.
      Le fichier `joueurVirtuel.c` contient une fonction qui donne un nombre d'allumettes à retirer en fonction de p et du nombre d'allumettes restantes. Utilisez cette fonction comme adversaire.

3. **La constante de Néper**

La constante de Néper est définie par :

$$e = \lim_{n \to \infty} \Sigma_{k=0}^{n} \frac{1}{k!}$$

La suite $u_n = \Sigma_{k=0}^{n} \frac{1}{k!}$ est une suite croissante qui tend vers e. Admettons que pour tout réel $\epsilon > 0$, si on a $u_{n+1} - u_n < \frac{\epsilon}{3}$ alors $e - u_{n+1} < \epsilon$. Programmer une fonction qui permet de calculer une approximation de e pour un ϵ donné. Par quoi est limité votre programme ?

4. **Les tours de Hanoï**

   Les tours de Hanoï est un jeu de réflexion consistant à déplacer des piles de disques de diamètres différents d'une tour de « départ » à une tour d'« arrivée » en passant par une tour « intermédiaire » et ceci en un minimum de coups, tout en respectant les règles suivantes :
   - on ne peut pas déplacer plus d'un disque à la fois,
   - on ne peut placer un disque que sur un autre disque plus grand que lui ou sur un emplacement vide.

   On suppose que cette dernière règle est également respectée dans la configuration de départ.

   Proposer un programmes qui demande à l'utilisateur un nombre de disques n et affiche la liste des déplacements à effectuer pour résoudre le problème comportant n disques.

   Par exemple pour n=2, le programme affichera :

```
Déplacer disque de la tour 1 vers la tour 2.
Déplacer disque de la tour 1 vers la tour 3.
Déplacer disque de la tour 2 vers la tour 3.
```

   Indice : on utilisera une fonction récursive dont le prototype est :

```
void hanoi(int n, int depart, int inter, int arrivee)
```