Intro au C++ M2 math Université Paris-Dauphine
denis.cornaz@lamsade.dauphine.fr

Un programme C++ est un ensemble de fonctions dont une `main()`. Elles peuvent être regroupées dans un ficher prog.cpp que l'on peut compiler par g++ prog.cpp produisant un exécutable a.out que l'on exécute en tapant la commande ./a.out.

Exercice 1 *Une approximation basique de $\sqrt{x}$ considère le rectangle $r \times \frac{x}{r}$. Si ce rectangle est un carré que vaut r ? Comment faire converger ce rectangle vers un carré ? Complétez, compiler et exécuter le code suivant:*

```
#include <iostream>
using namespace std;

double sqrt(int);

main(){
        cout <<"square root of 2 is approx." << sqrt(2)<<endl;
}
double sqrt(int n){
        double r=1.00;


        return r;
}
```

Exercice 2 *Complétez la fonction*

```
double Euler(int k){
        double r=1.00;


        return sqrt(6*r);
}
```

qui approxime π à l'aide de

$$\frac{1}{1^2} + \frac{1}{2^2} + \frac{1}{3^2} + \frac{1}{4^2} + \cdots + \frac{1}{k^2} + \cdots = \frac{\pi^2}{6}$$

Exercice 3 *Complétez la fonction*

```
double Walis(int it){
        double w=1.00;


        return w;
}
```

qui approxime π à l'aide de

$$\frac{\pi}{2} = \frac{2}{1} \cdot \frac{2}{3} \cdot \frac{4}{3} \cdot \frac{4}{5} \cdot \frac{6}{5} \cdot \frac{6}{7} \cdots$$

Exercice 4 *Soit x_0 un réel dans $[0,1]$. Le comportement de la "suite logistique"*

$$x_{n+1} = \mu \, x_n(1 - x_n)$$

est-il oscillatoire ? convergent ? chaotique ? Répondez en fonction du paramètre $\mu \in [0,4]$,

Exercice 5 *Pour a, b deux entiers, il existe un unique entier q et un unique entier r tels que $a = qb+r$ avec $0 \leq r < b$.*

1. *Implémentez* `int iquo(int a, int b)` *renvoyant q (équivalent de* `a/b`*) et* `int irem(int a, int b)` *renvoyant r (équivalent de* `a%b`*).*

2. *Implémentez une fonction* `int myst(int a, int b)` *qui multiplie a par deux, divise b par deux en arrondissant à l'entier inférieur, et retourne la somme des valeurs de a pour b impair. Par exemple avec*

$$\begin{array}{rcccc} a = & 11 & (22) & 44 & 88 \\ b = & 13 & 6 & 3 & 1 \end{array}$$

   elle retourne $11 + 44 + 88 = 143$.

3. *Que fait la fonction mystère ?*

Exercice 6

1. *Implémentez* `double fact(int n)` *retournant $n!$*

2. *Implémentez* `double pow(double x, int n)` *retournant x^n basée sur*

   (a) $x^n = x^{n-1} \times x$

   (b) $x^n = x^{\lfloor n/2 \rfloor} \, x^{\lceil n/2 \rceil}$

   Quelles sont les particularités de ces implémentations ?

3. *Implémentez* `double exp(double x, int it)` *retournant une valeur approchée de $\exp(x)$ en vous basant sur*

$$\exp(x) = \sum_{n=0}^{+\infty} \frac{x^n}{n!}$$

4. *Donnez une implémentation d'exponentiel en n'utilisant ni* `pow` *ni* `fact`*. Y-a-t il des changements d'efficacité ?*

5. *Trouver expérimentalement la valeur de $\lim_{x \to 0} \frac{\exp(x)-1}{x}$*

Exercice 7 *Complétez la fonction*

```
const int it=10;
double Brouncker(int i){
        if(i>it)
                return 0.00;
        if(i==0)
                return
        return
}
```

dont l'appel `Brouncker(0)` *approxime π à l'aide de*

$$\frac{4}{\pi} = 1 + \cfrac{1^2}{2 + \cfrac{3^2}{2 + \cfrac{5^2}{2 + \cfrac{7^2}{2 + \cfrac{9^2}{2 + \cfrac{11^2}{2 \; + \; ...}}}}}}$$

Exercice 8 *Une fonction réelle $f(x)$ est dérivable en x_0 si la limite*

$$f'(x_0) = \lim_{h \to 0} \frac{f(x_0 + h) - f(x_0)}{h}$$

existe. Trouver expérimentalement la valeur de b telle que $f(x) = b^x$ soit sa propre dérivée $f'(x_0) = f(x_0)$ en tout point $x_0 \in \mathbb{R}$.

Exercice 9 *Complétez la fonction*

```
const int it=10;
double pi(int i){
        if(i>it)
                return 0.00;
        if(i==0)
                return
        return
}
```

dont l'appel `pi(0)` *approxime π à l'aide de*

$$\pi = 3 + \cfrac{1^2}{6 + \cfrac{3^2}{6 + \cfrac{5^2}{6 + \ddots}}}$$

Exercice 10 *Implémentez une approximation de $\log_b(x)$.*

Exercice 11 *Complétez pour échanger :*

```
void f(int i, int j){
        void swap(int*,int*);
        cout<<i<<'\t'<<j<<endl;
        swap(&i,&j);
        cout<<i<<'\t'<<j<<endl;
}
void swap(int* pi,int* pj){
        *pi=*pi + *pj;


}
```

Exercice 12 *Complétez pour échanger :*

```
void f(int i, int j){

        cout<<i<<'\t'<<j<<endl;

        cout<<i<<'\t'<<j<<endl;
}
void swap(int & ri,int & rj){
        ri=ri + rj;


}
```

Exercice 13 *Complétez pour obtenir un programme affichant le triangle de Pascal.*

```
void pascal(int n){
  void affiche(int**,int);
  int** tri=new int*[n];
  for(int i=0;i<n;i++){


    for(int j=1;j<i;j++)

  }
  affiche(tri,n);
  for(int i=0;i<n;i++)
          delete[] tri[i];
  delete[] tri;
}
void affiche(int** tri, int n){
  for(int i=0;i<n;i++){
    cout << endl;
    for(int j=0;j<i+1;j++)
      cout << *( (*(tri+i))+j ) << "\t";
  }
}
```

Exercice 14 *Complétez ce programme pour qu'il affiche* $0, 1, \ldots, 2^n - 1$ *en base 2:*

```
#include <iostream>
using namespace std;

const int b=2;
const int n=3;
int T[n];
void print();
void compt(int);
main(){
        compt(0);
}
void print(){
        for(int i=0;i<n;i++)
                cout<<T[i]<<" ";
        cout<<endl;
}
void compt(int i){
        for (int j=0;j<b;j++){

                if(i==n-1)

                else

        }
}
```

Exercice 15 *Implémentez une fonction qui énumère toute les solutions entières du système linéaire* $Ax \leq \mathbf{1}$ *avec*

$$A = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{pmatrix} \quad \mathbf{1} = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

Exercice 16 1. *Donnez une fonction récursive* `int pgcd(int a, int b)` *retournant le plus grand commun diviseur de deux entiers.*

2. *Donnez une version itérative de* `int pgcd(int a, int b)`*.*

3. *Pour a, b deux entiers, le théorème de Bachet-Bezout assure l'existence de deux entiers relatifs u, v tels que $au + bv = d$, où $d =$pgcd(a, b). Complétez pour que la fonction* `f(a,b)` *les affiche.*

```
#include <iostream>
struct T{
        int d,u,v;

        void print(){std::cout<<d<<" "<<u<<" "<<v<<std::endl;}
};
T babez(int a, int b){
        if(b==0)
                return T(a,1,0);
        else{
                T t=babez(b,a%b);

        }
}
void f(int i, int j){

}
```

4. *Peut-il exister une fonction retournant deux entiers relatifs u, v différents mais avec la même propriété ?*

Exercice 17 *Implémentez une fonction qui énumère tous les placements possibles de 8 reines sur un échiquier sans aucune menace entre elles.*

On pourra faire un affichage rudimentaire de la solution, par exemple:

```
1) n=4
| | |Q| |
|Q| | | |
| | | |Q|
| |Q| | |

2) n=8
|Q| | | | | | | |
| | | | | | |Q| |
| | | | |Q| | | |
| | | | | | | |Q|
| |Q| | | | | | |
| | | |Q| | | | |
| | | | | |Q| | |
| | |Q| | | | | |
```