TP algo

1. Quelques calculs simples

$$e = \sum_{n=0}^{\infty} 1/n!:$$

```
float fact_div_T(int n){
    if (n==0){return 1.0;}
    else return (1.0/n)*fact_div_T(n-1);
}

float fact_div(int n){
    if(n==0){return fact_div_T(n);}
    else {return fact_div_T(n)+fact_div(n-1);}
}
```

Puissance

calculer (1 + 10−k)^10^k...

Sachant que (1 + x)1/x vaut environ
e ∗(1 −(1/2) ∗x + (11/24) ∗x2 −(7/16) ∗x3 + (2447/5760) ∗x4 −(959/2304) ∗x5 + O(x6))

	itératif	récursif	récursif terminale	récursif 2^p=2(n*2)
mémoire				
précision du résultat				
temps de calcul				

itératif

```
for (int i = 0;i<n;i++) a = a*x

for (int j = 0;j>n;j++) a = 1/a*1/x

cas 0 1
```

récursif

```
Puissance(x,n)

    if n == 0 then return 1

    if n>0 then x*puissance(x,n-1)

    else 1/x*puissance(x,n-1)
```

récursif terminal

```
puissance (x,n,acc)

    if n == 0 then return acc

    if n>0 then puissance(x,n-1,acc * x)

    else puissance(x,n-1, 1/x * acc)
```

récursif 2^p= 2(n*2)

```
pow(x,n)

    if n == 0 then return 1
    else if n>0
        if n%2==0 pow(x,n/2) * pow(x,n/2)
        else pow(x,n/2) * pow(x,n/2) * x
    else if n<0
        if n%2 == 0 pow(1/x,n/2) * pow(1/x,n/2)
        else pow(1/x,n/2) * pow(1/x,n/2) * 1/x
```

Ackermann

les premières valeurs de A(m,0)

itératif

```
int Ack(int m, int n){
    if (m == 0) {return n+1;}
    else res = 1;
        for(int i = 0;i<(m+1);i++) res = Ack(m-1,res);
    return res;
}
```

récursif

```
int Ackrec(int n, int m){
    if (m == 0){return n+1;}
    else if (n == 0){return Ack(m-1,1);}//ou m+1
    else return Ack(m-1, Ack(m,n-1);)

}
```

Suite

itératif

calculons x(100)

```
//La suite de réels (x(n))n∈N est définie par récurrence: x(0) = 1 puis ∀n ≥1,x(n) = x(n−1) +
1/x(n−1).
float suiteX(int n){//prend n et rend x(n)
    float res = 0;
    if(n == 0){res = 1;}
    else {
        for(int i = 0;i<n;i++){
            res = res + 1.0/res;
        }
    }
    return res;
}
```

récursif

```
float suiteXrec(int n){
    float res = 0;
    if (n == 0){return 1;}
    else {return suite(n-1)+(1.0/suite(n-1));}
}
```

2. Listes-Piles et Files

3. Arbres: Quadtrees