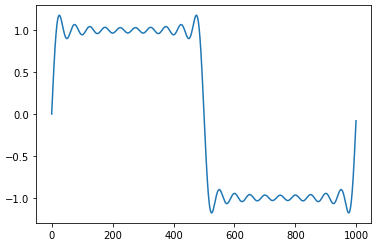
1. La période T0 de ce signal est de 2π. ω0 = 2π/T0 = 1. f0 = 1/T0 = 1/2π.
2. La fréquence d’échantillonnage doit être supérieure au double de la fréquence maximale de ce signal. Donc ici fs > 2.fmax 🡪 fs > 2.fn 🡪 fs > 2/Tn 🡪 fs > 2π.
3. Le signal s doit être périodique et respecter les conditions de Dirichlet suivante : le signal doit être continue par morceaux, monotone par morceaux et partout intégrable.

s(t) = 4/πsin(2πf0t) + 4/3πsin(2π(3f0)t) + 4/5πsin(2π(5f0)t) + …

1. Si l’on tronque la somme à un ordre n :

sn(t) =

1. Pour ce signal :

a0 = 4π.

ak = 0.

bk = 0 lorsque k est pair.

bk = 4/kπ lorsque k est impair.

1. En python on calcule un terme :

*terme = (2\*(1-(-1)\*\*k)\*np.sin(k\*x))/(k\*(np.pi));*

1. Seul le calcul du terme change :

**if**k%2 == 0: #Pair  
   terme = 2\*np.pi;  
**else**: #Impair  
   terme = (2\*np.pi)+(4/(k\*x\*np.pi))\*np.sin((2\*np.pi\*k\*x)\*2\*np.pi);

1. La convergence est « lente » (environ 1 / k). A partir de l’ordre N=79 on commence à observer un signal carré.
2. Il suffit de boucler de 1 à 100 par exemple et afficher le math.plot, on observer le résultat suivant :
3. On observe le phénomène de Gibbs. L’ « overshoot » est de 8.95 % de l’amplitude de la discontinuité, ce qui est négligeable lorsque k est grand.

