Homework 3

Jessica Perez

Parallel Computing Fall 2021

Question 1: Temperature Degree Conversion

Temperatures are measure by either Centigrade degress o Farenheit degress. The conversion equation between them is C/5 = F-32 / 9 where C is the Centigrade degree and F is the Fanhrenheit degree. You may use the equation to conver them

1. Write this two functions using loops to convert each temperaturue in a list one by one .
2. Write two new fucntions with Python Numpy to convert these temperatures without using loops
3. Compute the speed of these two fucntions and calculate the speedup

1. Create two functions to convert F to and C using sequential loops

In [32]:

```
def F2_C_loop(temps): #function for F 2 C loop
    C = np.zeros(len(temps))
    for i in range(len(temps)):
        C[i] = (5.0 * (temps[i]- 32.0))/9.0
        return C
def C2F_loop(temps):
    C = temps - 32
    return C
def Celcius2Floop(temps):
    return
```

In [33]:

```
%time F2_C_loop(F)
%time C2F_loop(F)
```

```
Wall time: 997 µs
Wall time: 2.99 ms
```

Out[33]:

```
array([25.18307563, -5.38636414, 43.20324352, ..., 53.76834564,
        5.09711082, 23.26403992])
```

2. Create functions with Python Numpy with out loops

In [42]:

```
def F2_C_loop(temps):
    C = 5 * (temps -32 )/9.0
    return C

def  C2F_loop(temps):
    return
```

In [47]:

```
%time F2FC(F)
%time C2F(F)
```

```
Wall time: 8.94 ms
Wall time: 0 ns
```

Speed Up

In [46]:

```
speedup = 5.4 *100/ 513
```

Question 2: Compute Euclidean distance using Numpy Arrays

1. Write a function using loop to compute the distance
2. Write a new function using Python Numpy to compute the distance with using loop
3. Compafe the speed up of these two fucntion and calculate the speedup

In [50]:

```
A = np.random.random(1000000)
B = np.random.random(1000000)
```

In [51]:

```
A-B
```

Out[51]:

```
array([ 0.08051054, -0.259559  ,  0.08994603, ..., -0.61000604,
        0.53131881, -0.50828201])
```

In [52]:

```
%time np.sqrt(np.sum((A-B)**2))
```

```
Wall time: 8.97 ms
```

Out[52]:

```
408.2335042013078
```

In [ ]: