Our project implementation has two parts:

a) Retiming techniques Implementation
b) Skewing the pivot

- All the programs are written in C++ and are executed on Hydra Machine

Retiming Techniques:

1) Open source code folder provided
2) Go to the sub-folder named retiming
3) In this folder, you have four files
    a) beforebubble: This c++ code is written to show the code of general bubble sort
    b) afterbubble: This c++ code is written to show you the bubble sortcode after we apply the iterative retiming technique
    c) beforeoddeven: This c++ code is written to show you the oddeven sort code before applying retiming
    d) afteroddeven: This c++ file is written to show you the oddeven sort code after applying the retiming techniques

NOTE: Execute each command atleast three times and take the average of the three values

Command On Hydra To Compile:

g++ -fopenmp filename.cpp -o filename

Command to Run The program:

bpsh 2 ./filename

Command To Calculate cpu-cycles:

perf stat -e cpu-cycles bpsh 1 ./filename

Command To Calculate Branch-misses:

perf stat -e branch-misses bpsh 1 ./filename

- **To calculate run time of the programs**
    1) Go to source code folder
    2) Go to runtime sub folder
    3) You can run the four source code files on any c++ IDE or C++ shell (available online tool) , it will return you the run time code as we have written code embedded for calculating runtime in these program files.

NOTE: Hydra do not support the clock() method we have used to calculate the run time of the programs, so we have used online c++ shell to calculate run time of our programs.

NOTE: Inputs are inbuilt

Skewing The Pivot:

For quick sort the position of pivot and type of input which we are providing will effect its run time and performance statistics.

Here, we used three kinds of input:

a) Ascending order input
b) Descending order input
c) Random order input

Positions of pivot:

a) Median element as pivot
b) First element as pivot
c) Last element as pivot
d) Random as pivot
e) Quicksort without recursion

Steps to run:

1) Go to source code folder
2) Go to sub-folder skewing the pivot
3) We have four sub-folders here:
    a) FirstElementAsPivot:
       This folder contains three files
        a) AscFirst: Ascending input and First element as pivot
        b) DscFirst: Descending input and First element as pivot
        c) RanFirst : Random input and first element as pivot
    b) LastElementsPivot:
       This folder contains three files
        a) asclast: Ascending input and last element as pivot
        b) dsclast: Descending input and last element as pivot
        c) ranlast: Random input and last element as pivot

    c) MedianAsPivot:
       This folder contains three files
        a) Median_A_50: Ascending input and median element as pivot
        b) Median_D_50: Descending input and median element as pivot
        c) Medain_R_50: Random input and median element as pivot

    d) RandomAsPivot:
       This folder contains three files

a) ascrandom: Ascending input and random element as pivot
b) dscrandom: Descending input and random element as pivot
c) QuickRandom: Random input and random element as pivot

e) Non-recursive: In this folder we have a file
Quickwithoutrecursion: This file gives you the quicksort code without recursion
Input: You can input any integer values, first input an integer which denotes the size of array ,suppose n and secondly input n integer values.

Command On Hydra To Compile:

g++ -fopenmp filename.cpp -o filename

Command to Run The program:

bpsh 2 ./filename

Command To Calculate cpu-cycles:

perf stat -e cpu-cycles bpsh 1 ./filename

Command To Calculate Branch-misses:

perf stat -e branch-misses bpsh 1 ./filename