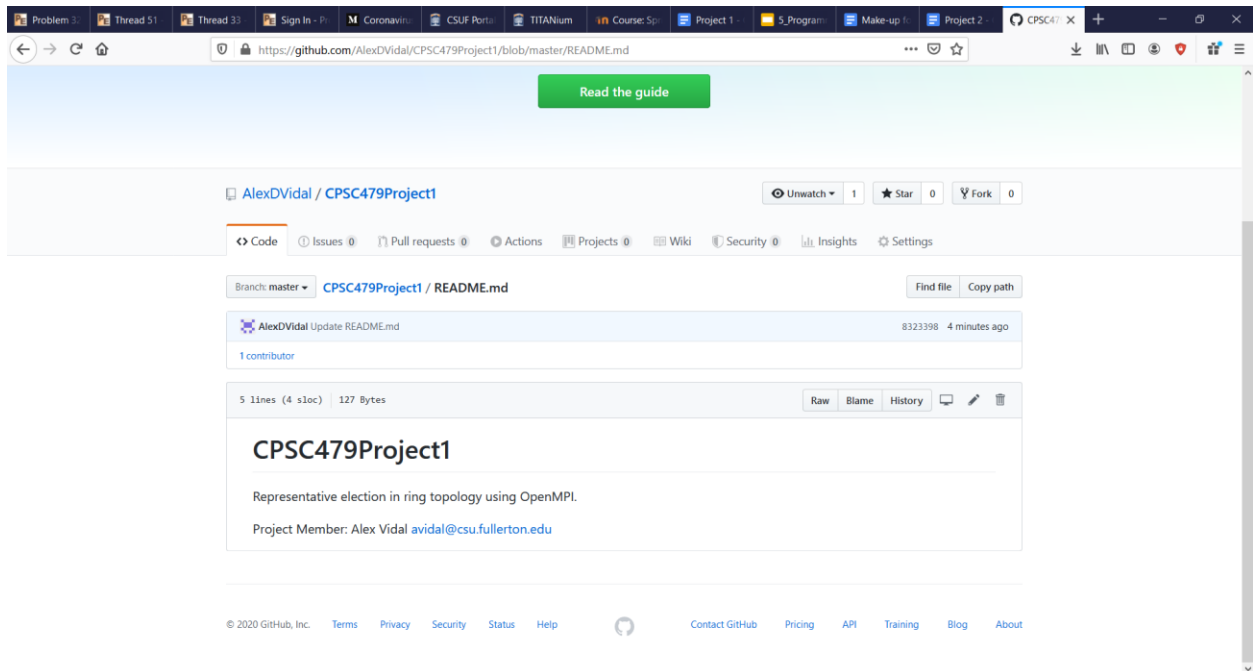


# Project 1 Representative Selection in a Ring Topology

Alex Vidal: [avidal@csu.fullerton.edu](mailto:avidal@csu.fullerton.edu)

## Screenshot of repository and members



## Pseudocode of the algorithm

```
//Find representative by means of passing generated numbers amongst
//neighbors in ring topology. If a process detects it has the smallest value of its
//group it will be considered the representative of the group.
int main(void){
    int rank, size, lowerRank, upperRank, randN, genN, lowerGen, upperGen
    string buffer

    //Start communications and find ranks
    MPI_Init()
    rank = MPI_Comm_Rank()
    size = MPI_Comm_Size()
    upperRank = (rank+1)%size
    if(rank == 0) {
        lowerRank = size-1 }
    else {
        lowerRank = rank - 1 }

    //Generate unique number
    seed(time())
    randN = rand() % 100
    if(randN < 10){ randN += 10 }

    //Generated number is a 1 followed by the random number
    //as 2 digits, and the rank as 2 digits. Rank ensure uniqueness
    sprintf(buffer, "1%02d%02d", randN, rank)
    sscanf(buffer, "%d", &genN)

    //Communicate generated numbers with neighbors. All communications are
    //non-blocking to simplify code. Synchronize with a wait
    MPI_Irecv(lowerGen, lowerRank)
    MPI_Irecv(upperGen, upperRank)
    MPI_Isend(genN, lowerRank)
    MPI_Isend(genN, upperRank)
    MPI_Wait()

    //print details and if representative
    printf("Rank %d, rand %d, gen %d, lower %d, upper %d. ",
        rank, randN, genN, lowerGen, upperGen)
    if(genN < lowerGen && genN < upperGen){
        printf("I am representative.")
    }
    printf("\n");

    //close comms
    MPI_Finalize()
}
```

## Usage

To build the project type the following command, it will create an executable named “project1”:

```
~$ mpicc project.c -o project1
```

Once the project builds you can run it with the following command:

```
~$ mpirun -n <X> ./project1
```

Where <X> can be any integer less than 100, but for the purposes of this project is between 6 and 20. Here’s some sample output.

```
[avidal@titanv makeupP1]$ mpirun -n 7 ./project1
Rank:  2 Rand: 62 Gen: 16202 Lower: 19501 Upper: 13003.
Rank:  3 Rand: 30 Gen: 13003 Lower: 16202 Upper: 17004. I am representative.
Rank:  4 Rand: 70 Gen: 17004 Lower: 13003 Upper: 11605.
Rank:  5 Rand: 16 Gen: 11605 Lower: 17004 Upper: 11506.
Rank:  0 Rand: 12 Gen: 11200 Lower: 11506 Upper: 19501. I am representative.
Rank:  1 Rand: 95 Gen: 19501 Lower: 11200 Upper: 16202.
Rank:  6 Rand: 15 Gen: 11506 Lower: 11605 Upper: 11200.
[avidal@titanv makeupP1]$
```

Each row of output is a single process printing its rank and the generated numbers of itself and its neighbors. If it has the lowest number of the group it will additionally print that it is a representative.

## Output screenshots

Here are screenshots of two different runs:

15 processes resulting in 4 representatives

```
avidal@titanv:~/CPSC479/makeupP1
[avidal@titanv makeupP1]$ mpirun -n 15 ./project1
Rank: 1 Rand: 19 Gen: 11901 Lower: 16300 Upper: 16302. I am representative.
Rank: 2 Rand: 63 Gen: 16302 Lower: 11901 Upper: 12403.
Rank: 3 Rand: 24 Gen: 12403 Lower: 16302 Upper: 19704. I am representative.
Rank: 4 Rand: 97 Gen: 19704 Lower: 12403 Upper: 17405.
Rank: 5 Rand: 74 Gen: 17405 Lower: 19704 Upper: 13306.
Rank: 6 Rand: 33 Gen: 13306 Lower: 17405 Upper: 11507.
Rank: 7 Rand: 15 Gen: 11507 Lower: 13306 Upper: 16208. I am representative.
Rank: 8 Rand: 62 Gen: 16208 Lower: 11507 Upper: 18409.
Rank: 9 Rand: 84 Gen: 18409 Lower: 16208 Upper: 19510.
Rank: 10 Rand: 95 Gen: 19510 Lower: 18409 Upper: 17411.
Rank: 11 Rand: 74 Gen: 17411 Lower: 19510 Upper: 16312.
Rank: 12 Rand: 63 Gen: 16312 Lower: 17411 Upper: 16113.
Rank: 13 Rand: 61 Gen: 16113 Lower: 16312 Upper: 18614. I am representative.
Rank: 14 Rand: 86 Gen: 18614 Lower: 16113 Upper: 16300.
Rank: 0 Rand: 63 Gen: 16300 Lower: 18614 Upper: 11901.
[avidal@titanv makeupP1]$
```

20 processes resulting in 7 representatives

```
avidal@titanv:~/CPSC479/makeupP1
[avidal@titanv makeupP1]$ mpirun -n 20 ./project1
Rank: 1 Rand: 16 Gen: 11601 Lower: 11400 Upper: 14702.
Rank: 2 Rand: 47 Gen: 14702 Lower: 11601 Upper: 13703.
Rank: 3 Rand: 37 Gen: 13703 Lower: 14702 Upper: 15904. I am representative.
Rank: 4 Rand: 59 Gen: 15904 Lower: 13703 Upper: 11005.
Rank: 5 Rand: 10 Gen: 11005 Lower: 15904 Upper: 11906. I am representative.
Rank: 8 Rand: 77 Gen: 17708 Lower: 16507 Upper: 11109.
Rank: 18 Rand: 41 Gen: 14118 Lower: 19017 Upper: 16519. I am representative.
Rank: 19 Rand: 65 Gen: 16519 Lower: 14118 Upper: 11400.
Rank: 6 Rand: 19 Gen: 11906 Lower: 11005 Upper: 16507.
Rank: 7 Rand: 65 Gen: 16507 Lower: 11906 Upper: 17708.
Rank: 9 Rand: 11 Gen: 11109 Lower: 17708 Upper: 18010. I am representative.
Rank: 10 Rand: 80 Gen: 18010 Lower: 11109 Upper: 11811.
Rank: 11 Rand: 18 Gen: 11811 Lower: 18010 Upper: 13012. I am representative.
Rank: 12 Rand: 30 Gen: 13012 Lower: 11811 Upper: 15413.
Rank: 13 Rand: 54 Gen: 15413 Lower: 13012 Upper: 11614.
Rank: 17 Rand: 90 Gen: 19017 Lower: 18616 Upper: 14118.
Rank: 0 Rand: 14 Gen: 11400 Lower: 16519 Upper: 11601. I am representative.
Rank: 14 Rand: 16 Gen: 11614 Lower: 15413 Upper: 11315.
Rank: 15 Rand: 13 Gen: 11315 Lower: 11614 Upper: 18616. I am representative.
Rank: 16 Rand: 86 Gen: 18616 Lower: 11315 Upper: 19017.
[avidal@titanv makeupP1]$
```