

Day #6 Exercise – SOLUTION

Compute jobs arrive to a compute server. Your task is to model the CPU demand of the jobs. You have made measurements on 1000 jobs and found that:

Job size in milliseconds (CPU demand)	Number of jobs
1	100
2	300
3	100
4	300
5	200

Write a C function that will return a simulated job size for an empirical distribution based on the above measurements – that is, write a workload simulation. Assume that you have a function `rand_val()` available to use that returns a uniformly distributed random value (of type double) between 0.0 and 1.0.

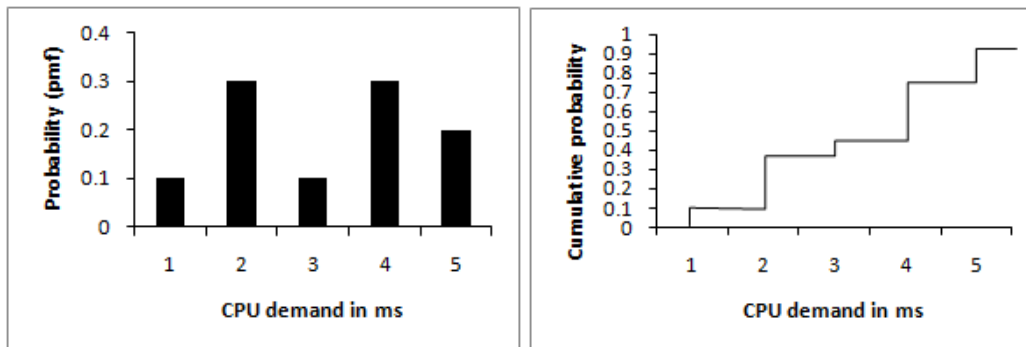
```
int jobSize()
{
    double z;

    z = rand_val();

    if (z <= (double) 100 / 1000) return(1);
    if (z <= (double) 400 / 1000) return(2);
    if (z <= (double) 500 / 1000) return(3);
    if (z <= (double) 800 / 1000) return(4);
    return(5);
}
```

Answer the following questions regarding your workload simulation:

a) Sketch the pmf and CDF of the job size empirical distribution.



b) Will the job sizes generated by your function be independent of each other? Why or why not?

The job sizes will be independent of each other as long as the underlying random number generator generates random uniform RVs.

c) If you wanted to generate job sizes that were not independent (that is, they would have a given autocorrelation value of non-zero for some lags), how would you do so?

This is a tough question. I don't know the answer – I don't think that there is a simple answer. If an underlying sequence of uniform RVs that has the desired autocorrelation sequence could be generated, this would be a first step.

d) Suppose you expect that in the future the number of large jobs (jobs with 5 millisecond CPU demand) will double, how would your function change? Make the change.

The number of jobs with 5 milliseconds demand would now be 400 for a total of 1200 jobs. This changes the code as follows:

```
if (z <= (double) 100 / 1200) return(1);  
if (z <= (double) 400 / 1200) return(2);  
if (z <= (double) 500 / 1200) return(3);  
if (z <= (double) 800 / 1200) return(4);  
return(5);
```