```
/*****************************************************************************
* File: goodputRatio.cpp
* Author: Ian Downard
* Date: 20 Apr 2004
* Description:
*   Same as goodputRatio.cpp, except it reads in data for simulations with two
* sources
*
* COMPILE:
*
*  g++ goodputRatio2.cpp -Wno-deprecated -o goodputRatio2
*
* USAGE:
*
*   ./goodputRatio maxGoodput sample.dat
*
*   Goodput samples will be read from time 0 thru the end of the dat file.
*     Dat files are prodecued as shown in the usage example.
*   Goodput will be computed as a percentage of the maximum theoretical
*     goodput, as specified in the _maxGoodput_ parameter.
*
* USAGE EXAMPLE:
*
*   ns sample.tcl
*     stripDups sample-n1.drc > sample-n1-nodups
*     stripDups sample-n2.drc > sample-n2-nodups
*     stripDups sample-n3.drc > sample-n3-nodups
*     stripDups sample-n4.drc > sample-n4-nodups
*     stripDups sample-n5.drc > sample-n5-nodups
*   cat sample-n{1,2,3,4,5}-nodups > here.drc
*   sort here.drc > sample.drc
*   trpr drec input sample.drc recv window 5 output sample.pl
*   grep ", " sample.pl > sample.dat
*   ./goodputRatio 590 250 sample.dat
*
*****************************************************************************/

#include <fstream.h>
#include <string.h>

void Usage();

int main(int argc, char* argv[])
{
  if(argc != 3)
  {
    Usage();
    return 1;
  }

  FILE *fp = fopen(argv[2], "r");
  if (fp == NULL)
  {
    fprintf(stderr, "%s: can't open %s\n", argv[0], argv[1]);
    return(1);
  }

  int  lines = 0;
  int previous_sample_time = -1;
  float goodputSum, percentage, goodputAvg;
  float goodput, goodput1, timestamp = 0.0;
  float maxGoodput = atol(argv[1]);

  while ((fscanf(fp, "%f, %f, %f", &timestamp,  &goodput, &goodput1)) != EOF)
  {
    if (int(timestamp) != previous_sample_time) {
//      fprintf(stderr, "%f, %d\n", timestamp, previous_sample_time);
      fprintf(stderr, "Read %f, %f, %f\n", timestamp, goodput, goodput1);
      goodputSum += goodput + goodput1;
```



```
      lines ++;
      goodputAvg = goodputSum / lines;
      percentage = goodputAvg / maxGoodput * 100;
    }
    previous_sample_time = int(timestamp);
  }

/*
  printf("Average Cumulative Goodput %4.2f kbps is %2.2f%% of the\ntheoretical m
axGoodputimum %4.2f kbps.\n",
    goodputAvg, percentage, maxGoodput);
*/
  printf("%2.2f%%\n", percentage);

  fclose(fp);
  return 0;

}

void Usage()
{
  fprintf(stderr, "Missing required arguments!\n");
  fprintf(stderr, "USAGE: goodputRatio2 max_goodput(kbps) goodput.data\n\n");
  fprintf(stderr, "Example: ./goodputRatio2 250 sample.dat\n\n");
  return;
}
```