

METU – Dept. of Computer Engineering - CENG315
Fall 2013-2014 -- Homework #2
(Deadline: 24th November 2013, Sunday)

In this programming homework you are going to generate an algorithm for a given problem using different sorting algorithms that will be specified. Here is the problem;

There is going to be a Grand Ant Race soon. It is estimated N ants to participate in the race. The racecourse is a straight line which is 100 metres long. All of the ants are going to start the race on the same level and at the same time. The speeds of the ants can be different as well as some of them can be the same. The race consists of 5 laps and the speed of an ant can be different for each lap. The speeds will be in metres/second. Since the race takes a long time, the ants need to have a rest during each lap. Similarly, each ants' time of resting may be different from each other's, as well as they may be the same. Also, similarly, each ants' time of resting may be different at each lap. The times of resting will be in seconds. After each lap, the ants are going to be pointed according to their ranks. The points are going to be in the range between 0 and N (the number of ants). The ant(s) which completes a lap as the first will take N points and the other ants will take the points in decreasing order (decreased by 1) with respect to their ranks. To illustrate, the first(s) will take N points, the second(s) will take $N-1$ points, the third(s) will take $N-2$ points, ... etc. Since an ant may have different speed for each lap, the points that it takes in the end of each lap may differ. At the last of the race, the points of the 5 laps are going to be summed for each ant and the ant(s) who takes the highest point will be the first of the race.

Your duty is to calculate the points of the ants for each lap and then print the total points of the ants that they accumulated at the end of the race, in decreasing order.

Specifications of the Homework

Implement all the following sorting algorithms in your homework. You have to solve the given homework problem using them:

- 1) Well-tuned Quick Sorting (with median of 3 partitioning)
- 2) Shell Sort (take "gap" as "powers of two minus one": 1, 3, 7, 15, 31, 63, ...)
- 3) Bucket Sort (you can divide the speed interval to 10 equal parts)
- 4) Radix Sort

Also, you need to prepare a written report, including the charts comparing the running times of the sorting algorithms vs N .

You should implement in C/C++.

INPUT Specifications

Your program is to read from a file named "hw2.inp" where the first line represents the type of sorting algorithm(1,2,3,4 as above), the second line represents the number of ants (N), and then (beginning from the third line) at each line an ant's speeds and times of resting respecting to the 5 laps are going to be shown. This representation will be in the order that: speed in the 1st lap, 1 character space, speed in the 2nd lap,, 1 character space, speed in the 5th lap, 1 character space, time of resting in the 1st lap, 1 character space, time of resting in the 2nd lap,

Sample "hw2.inp" :

```
3
4
0.1  0.5  0.9  0.23 0.12 145 23  12 40  160
0.15 0.2  0.15 0.2  0.1  60  150 60 104 150
0.5  1.0  0.7  0.7  0.1  27  123 20 28  144
0.32 0.15 0.17 0.2  0.24 60  100 81 120 81
```

(Here, the spaces between the speeds(simirlarly times of resting) are more than 1-character-space to be readable, but while you're implementing your code, you will assume that there will be only 1-character-space between the speeds(and times of resting) in the inputs)

OUTPUT Specifications

Your code should generate a file named "hw2.out". As the first 6 lines, you should output the execution time of the specified sorting algorithm. The execution time that was found in each step (in each Lap and total points) must be printed in milliseconds. Then in the following lines, the id of the ants together with the total point of the ants will be printed in decreasing order with respect to the points. Use space as a delimiter in each line. (The id of an ant is actually the order in which the ant was defined in the input)

The output example below belongs to the input example above:

Sample "hw2.out" :

```
time_of_bucketsorting      (To sort Lap1)
time_of_bucketsorting      (To sort Lap2)
time_of_bucketsorting      (To sort Lap3)
time_of_bucketsorting      (To sort Lap4)
time_of_bucketsorting      (To sort Lap5)
time_of_bucketsorting      (To sort the total points)
17 3
15 1
12 4
9 2
```

First line shows the time required for Bucket Sort since 3 is given in input.

For the other lines:

When it is calculated we see that times spent for each ant are nearly:

1 --> 1145 223 123 474 993
2 --> 726 650 726 604 1150
3 --> 227 223 162 170 1144
4 --> 372 776 669 620 497

Hence the points accumulated are:

1 --> $1 + 4 + 4 + 3 + 3 = 15$
2 --> $2 + 3 + 1 + 2 + 1 = 9$
3 --> $4 + 4 + 3 + 4 + 2 = 17$
4 --> $3 + 2 + 2 + 1 + 4 = 12$

Written part

You are expected to bring in a short report. Examine each sorting algorithm for the arrays having {50, 100, 500, 1000, 5000, 10000} elements(N). Then, plot a chart with axes **running time of sorting algorithms** vs **N** (When plotting, run the algorithms on the same input) for each sorting method you use. Plot those (4) charts on one single chart (using Excel or OpenOffice Calc) and add a paragraph explaining the details (*both explanation and pseudocode*) of your **sorting algorithm** in your report. In explanation you are expected to specify the best-worst-average cases of the 4 algorithms by saying the time complexities and for which inputs there occurs best-worst-average cases.

Sample code for time measurement

Please use this code for calculating the running times.

```
#include <sys/time.h>
...
struct timeval tv_begin;
struct timeval tv_end;
int passed_milliseconds;
gettimeofday(&tv_begin, NULL);
sort();
gettimeofday(&tv_end, NULL);
passed_milliseconds = (tv_end.tv_sec - tv_begin.tv_sec)*1000 + (tv_end.tv_usec - tv_begin.tv_usec)/1000;
```

Submission

- ☐ Upload hw2.c or hw2.cpp files to COW until 24th of November, 23:59. Bring the written report to A-310 until 17:00 the next day (25th of Nov. , Monday).
- ☐ All work should be done individually. In case you cheat, you will get 0 from your homework and disciplinary action will be taken.