

Algorithm Analysis Practice Exam. December 2020

Student: Pablo Cuesta Sierra

Theory Group: 125

Practice group: 1251

1. Check by plotting a histogram that the previous generator follows a linear distribution.

Using exercise1.c to use this function, we get the following histogram: (we can see that the probability increases linearly)

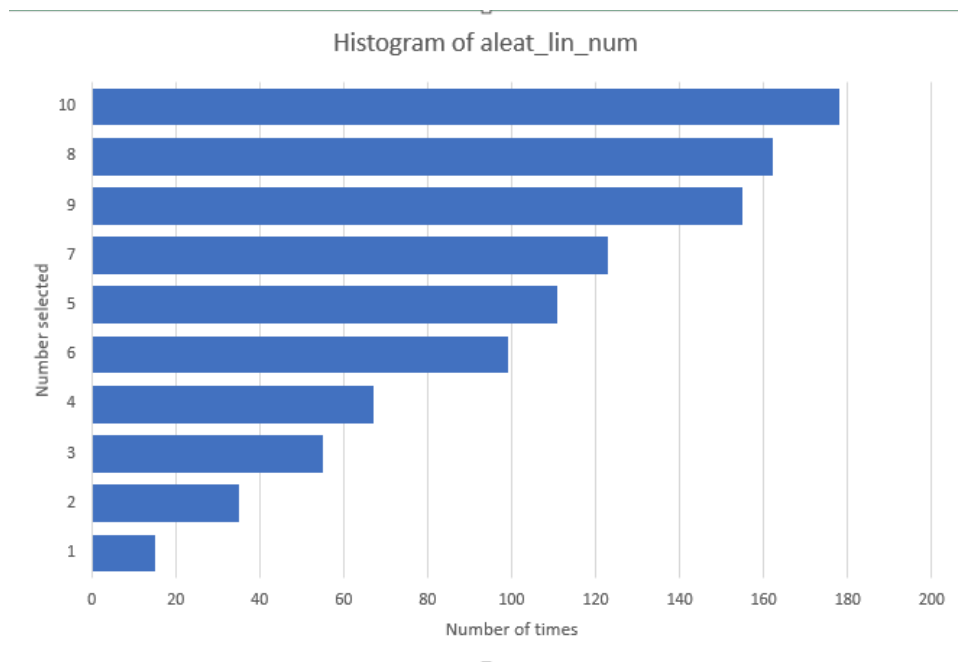


Figure 1

2. Implement bubble sort algorithm.

```
/*bubble sort*/
int bubblesort(int* table, int F, int L){
    int i, j, B0=0;

    if(F > L || table==NULL){
        return ERR;
    }

    for(i=L;i>F;i--){
        for(j=F;j<i;j++){
            B0++;
            if(table[j]>table[j+1]){
                swap(table+j, table+j+1);
            }
        }
    }
    return B0;
}
```

Figure 2

3. Measure and plot in the same figure the BO required to sort (using bubble sort) permutations generated using `aleat_lin_num()` and `aleat_num()` to compare the results.

We need to introduce in `permutations.c` these two new functions, to generate using `aleat_lin_num()`:

```
int* generate_perm_lin(int N){
    int *perm=NULL;
    int i;

    if(N<=0)
        return NULL;

    if(!(perm=(int*)calloc(N,sizeof(int))))
        return NULL;

    for(i=0;i<N;i++){
        perm[i]=aleat_lin_num(1,N);
    }

    return perm;
}

int** generate_permutations_lin(int n_perms, int N)
{
    int **perm=NULL;
    int i,j;

    if(n_perms<=0||N<=0)
        return NULL;

    if(!(perm=(int**)calloc(n_perms,sizeof(int*))))
        return NULL;

    for(i=0;i<n_perms;i++){
        perm[i]=generate_perm_lin(N);
        if(!perm[i]){
            for(j=0;j<i;j++)
                free(perm[j]);
            free(perm);
            return NULL;
        }
    }

    return perm;
}
```

Figure 3

And in `average_sorting_times`, we have to change `generate_permutations()` by `generate_permutations_lin()`.

For the experiments I have used the parameters provided (BubbleSort is quite slow, so higher sizes would require a lot of time with my computer).

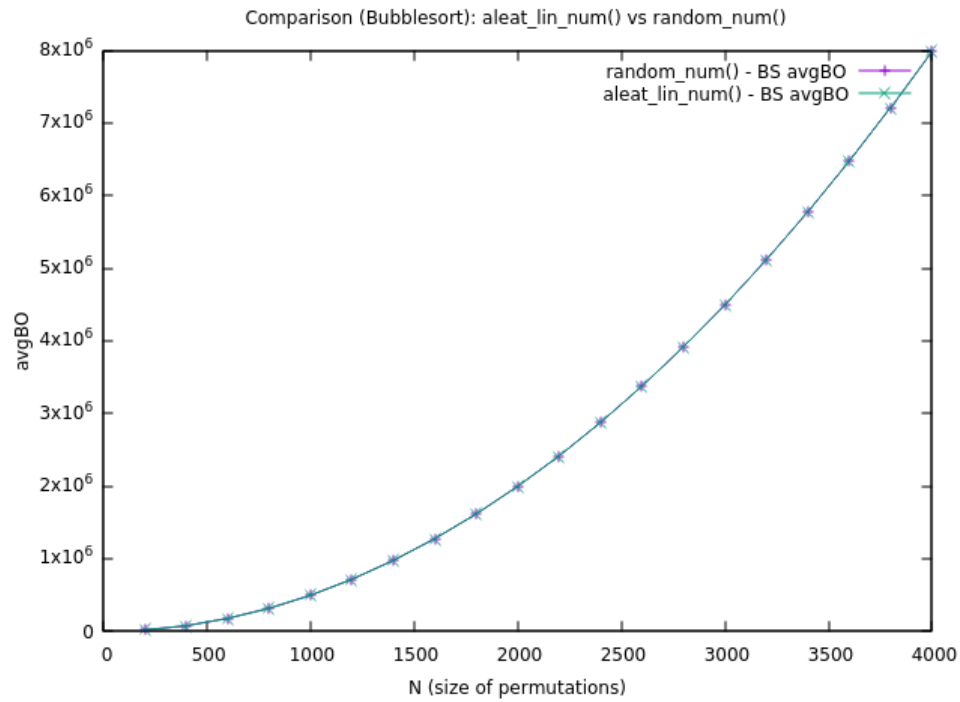


Figure 4

4. Discuss the result.

As we can see, there are no changes between the two different generations. This is because bubble sort performs the same operations regardless of the initial table, it has a cost, for a table of size N, of $N(N-1)/2 \sim N^2/2$. Therefore, the result is exactly the same.