**UNIVERSIDAD NACIONAL DE SAN AGUSTÍN DE AREQUIPA**

**FACULTAD DE PRODUCCIÓN Y SERVICIOS**

**ESCUELA PROFESIONAL DE CIENCIA DE LA COMPUTACIÓN**

**ANÁLISIS Y DISEÑO DE ALGORITMOS**

**GRUPO: B SEMESTRE: 2022 - A**

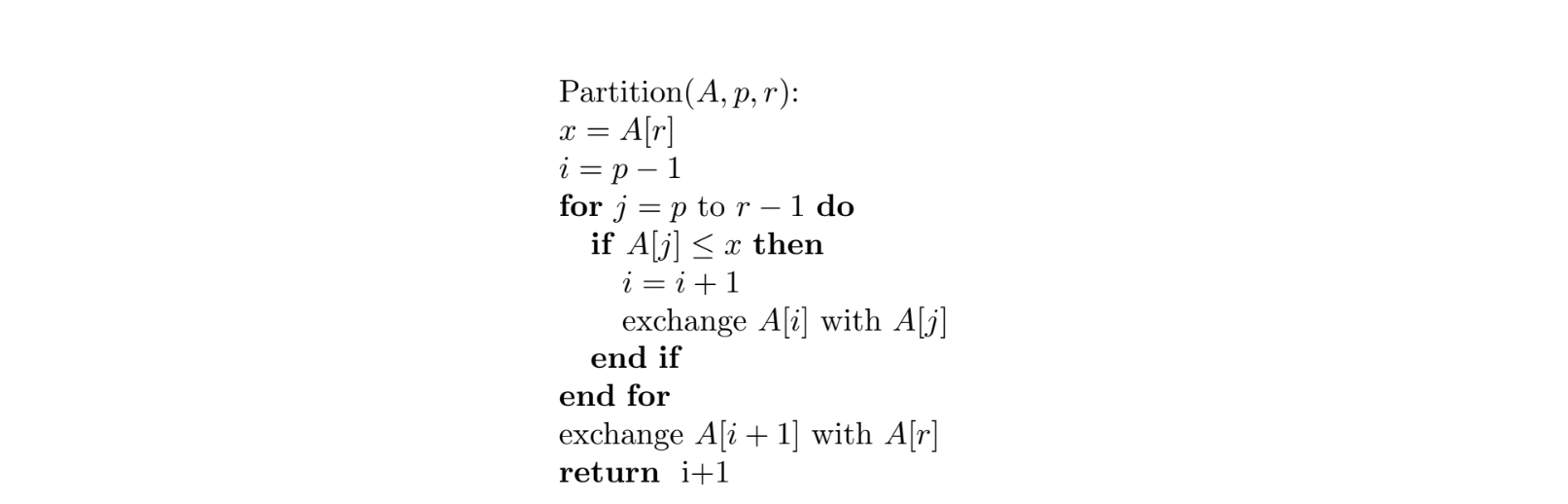
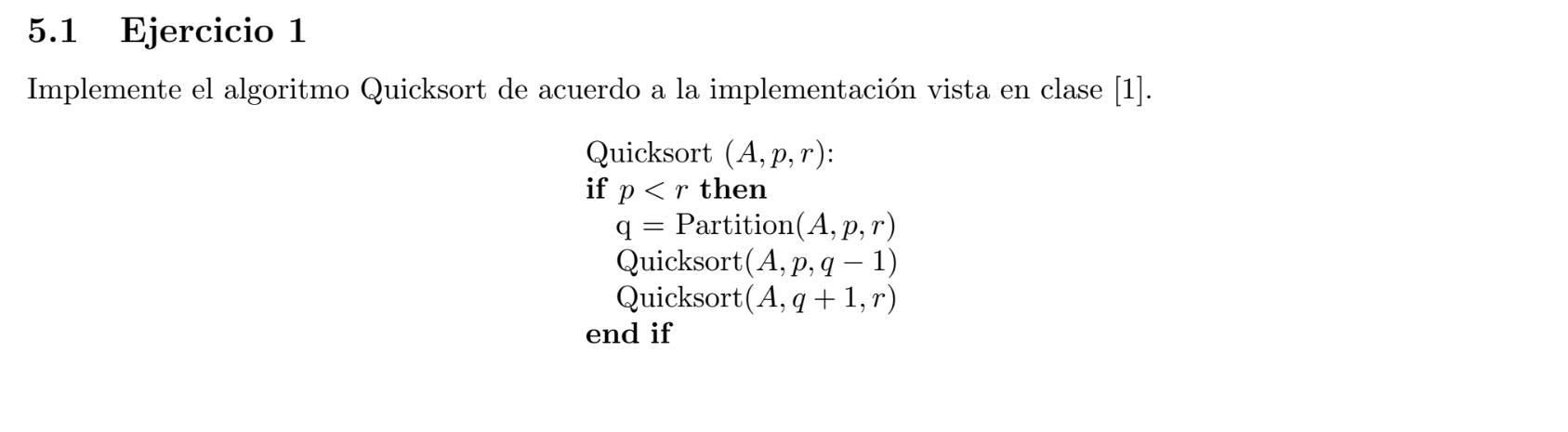
**INFORME DE LABORATORIO 6**

**DOCENTE: ROLANDO JESUS CARDENAS TALAVERA**

**ESTUDIANTE: GABRIEL PACCO HUARACA**

**Arequipa – Perú**

**2022**



#include <iostream>

#include <chrono>

#include <time.h>

using namespace std;

struct Timer

{

std::chrono::system\_clock::time\_point m\_start;

Timer()

{

    m\_start = std::chrono::system\_clock::now();

}

~Timer()

{

    auto end = std::chrono::system\_clock::now();

    std::chrono::duration<float, std::milli> duration = end - m\_start;

    std::cout << duration.count() << " milisegundos" << std::endl;

}};

*// Generar numeros aleatorios*

void generarNumeros(int \*A, int n)

{

    srand((unsigned int)time(NULL));

    for (int i = 0; i < n; i++)

    {

        A[i] = rand() % 10000;

    }

}

int Partition(int \*A, int p, int r)

{

    int pivot = A[r];

    int i = p - 1;

    for (int j = p; j < r; j++)

    {

        if (A[j] <= pivot)

        {

            i++;

            swap(A[i], A[j]);

        }

    }

    swap(A[i + 1], A[r]);

    return i + 1;

}

void QuickSort(int \*A, int p, int r)

{

    if (p < r)

    {

        int pivot = Partition(A, p, r);

        QuickSort(A, p, pivot - 1);

        QuickSort(A, pivot + 1, r);

    }

}

int main()

{

    int \*A;

    int n = 10;

    A = new int[n];

*//generarNumeros(A, n);*

    A = new int [n] { 6, 7, 8, 9, 10, 1, 2, 3, 4, 5};

    QuickSort(A, 0, n - 1);

    Timer t;

    for (int i = 0; i < n; i++)

    {

        cout << A[i] << " ";

    }

    cout << endl;

    return 0;

}

