OPERATING SYSTEM LAB

LAB QUIZ NUMBER 02

MUHAMMAD ABDULLAH | 221546 | BSCYS – IV – A

```
#include <iostream>
using namespace std;

int main()
{
    int A[100][4];
    int i, j, n, total = 0, index, temp;
    float avg_wt;

    cout << "ENTER NUMBER OF PROCESSES: ";
    cin >> n;
    cout << "-----------------------------------" << endl;

    cout << "ENTER BURST TIME: " << endl;

    for (i = 0; i < n; i++) {
        cout << "P" << i + 1 << ": ";
        cin >> A[i][1];
        A[i][0] = i + 1;
    }
    cout << "-----------------------------------" << endl;

    for (i = 0; i < n; i++) {
        index = i;
        for (j = i + 1; j < n; j++)
            if (A[j][1] < A[index][1])
                index = j;
        temp = A[i][1];
        A[i][1] = A[index][1];
        A[index][1] = temp;

        temp = A[i][0];
        A[i][0] = A[index][0];
        A[index][0] = temp;
    }

    A[0][2] = 0;
    for (i = 1; i < n; i++) {
        A[i][2] = 0;
        for (j = 0; j < i; j++)
            A[i][2] += A[j][1];
        total += A[i][2];
    }

    avg_wt = (float)total / n;
```

```
    total = 0;
    cout << "P     BT     WT" << endl;
    cout << "-----------------------------------" << endl;

    for (i = 0; i < n; i++) {
        A[i][3] = A[i][1] + A[i][2];
        total += A[i][3];
        cout << "P" << A[i][0] << "     " << A[i][1] << "     " <<
A[i][2] << endl;
    }

    cout << "-----------------------------------" << endl;
    cout << "AVERAGE WAITING TIME: " << avg_wt << endl;
    cout << "-----------------------------------" << endl;
}
```

OUTPUT:

```
ENTER NUMBER OF PROCESSES: 5
-----------------------------------
ENTER BURST TIME:
P1: 5
P2: 3
P3: 8
P4: 2
P5: 6
-----------------------------------
P     BT     WT
-----------------------------------
P4     2     0
P2     3     2
P1     5     5
P5     6     10
P3     8     16
-----------------------------------
AVERAGE WAITING TIME: 6.6
-----------------------------------

--------------------------------
Process exited after 15.94 seconds with return value 0
Press any key to continue . . .
```

CODE EXPLANATION:

This C++ code simulates a basic scheduling algorithm to manage processes' execution by calculating their average waiting time. Initially, it prompts the user to input the number of processes and their corresponding burst times. Subsequently, it employs the selection sort algorithm to arrange the processes in ascending order based on their burst times. After sorting, it iteratively computes the waiting time for each process, which accumulates the sum of burst times of all preceding processes. By summing these waiting times, it derives the total waiting time, dividing it by the number of processes to obtain the average waiting time. The code concludes by displaying a tabulated view of each process's ID, burst time, and waiting time, followed by the computed average waiting time. This process scheduling approach offers insights into optimizing system performance and resource allocation.

****