Q.2(a)

```c
#include <stdio.h>
int main()
{
    int i, n, sum = 0, count = 0, y, q, wt = 0, tat = 0, at[10], bt[10],
temp[10];
    float avg_wt, avg_tat;
    printf("Total number of process :");
    scanf("%d", &n);
    y = n;
    for (i = 0; i < n; i++)
    {
        printf("\n Please type the Arrival and Burst time of the
Process[%d]\n", i + 1);
        printf(" Arrival time is: \t");
        scanf("%d", &at[i]);
        printf(" \nBurst time is: \t");
        scanf("%d", &bt[i]);
        temp[i] = bt[i];
    }
    printf("Type the Time Quantum for the process: \t");
    scanf("%d", &q);

    printf("\n Process No \t\t Burst Time \t\t TAT \t\t Waiting Time ");
    for (sum = 0, i = 0; y != 0;)
    {
        if (temp[i] <= q && temp[i] > 0)
        {
            sum = sum + temp[i];
            temp[i] = 0;
            count = 1;
        }
        else if (temp[i] > 0)
        {
            temp[i] = temp[i] - q;
            sum = sum + q;
        }
        if (temp[i] == 0 && count == 1)
        {
            y--;
            printf("\nProcess No[%d] \t\t %d\t\t\t\t %d\t\t\t %d", i + 1,
bt[i], sum - at[i], sum - at[i] - bt[i]);
            wt = wt + sum - at[i] - bt[i];
            tat = tat + sum - at[i];
            count = 0;
```

```
        }
        if (i == n - 1) {
            i = 0;
        }
        else if (at[i + 1] <= sum) {
            i++;
        }
        else {
            i = 0;
        }
    }
    avg_wt = wt * 1.0 / n;
    avg_tat = tat * 1.0 / n;
    printf("\n Average Turn Around Time Is : \t%f", avg_wt);
    printf("\n Average Waiting Time Is: \t%f", avg_tat);
    getch();
}
```

Q. (3)

```c
#include <stdio.h>
#include <stdlib.h>
int mutex = 1;
int total_slots = 0;
int empty_slots = 10;
int m = 0;
void producer()
{
    --mutex;
    ++total_slots;
    --empty_slots;
    m++;
    printf("\nProducer consumes item : %d",m);
    ++mutex;
}
void consumer()
{
    --mutex;
    --total_slots;
    ++empty_slots;
    printf("\nConsumer consumes item : %d",m);
    m--;
    ++mutex;
}
int main()
{
    int n, i;
    printf("\n1. Press 1 for Producer to produce"
```

```c
        "\n2. Press 2 for Consumer to consume"
        "\n3. Press 3 for Exit");
#pragma omp critical
    for (i = 1; i > 0; i++) {

        printf("\nEnter your choice:");
        scanf("%d", &n);
        switch (n) {
        case 1:
            if ((mutex == 1)
                && (empty_slots != 0)) {
                producer();
            }

            else {
                printf("Buffer is full!");
            }
            break;

        case 2:
            if ((mutex == 1)
                && (total_slots != 0)) {
                consumer();
            }
            else {
                printf("Buffer is empty!");
            }
            break;
        case 3:
            exit(0);
            break;
        }
    }
}
```