



Question 01:- Implement algorithm to perform arithmetic operations b/w two numbers and numbers is input by the keyboard:

Answer 01:-

Code:-

```
#include <stdio.h>
int main () {
    int num1, num2;
    printf("Enter the first number: ");
    scanf("%d", &num1);
    printf("Enter the second number: ");
    scanf("%d", &num2);
    printf("Sum of entered number is: %d \n", num1+num2);
    printf("Difference between the entered number is %d \n",
    num1-num2);
    printf("product of entered number is %d \n" num1*num2);
    printf("On perform division we get quotient is %d \n" num1/num2);
    return 0;
}
```

Output:- Enter the first number: 23

Enter the second number: 7

Sum of entered number is: 30

Difference between the entered number is 16

Product of entered number is 161

On perform division we get quotient is 3



Question 02:- Implement algorithm to perform the swapping of two number by the using of third variable.

Answer 02:-

Codes:-

```
#include <stdio.h>
```

```
int main () {
```

```
    int num1, num2, temp;
```

```
    printf("Enter the value of number 1: ");
```

```
    scanf("%d", &num1);
```

```
    printf("Enter the value of number 2: ");
```

```
    scanf("%d", &num2);
```

```
    temp = num1;
```

```
    num1 = num2;
```

```
    num2 = temp;
```

```
    printf("The value of number 1 is %d\n", num1);
```

```
    printf("The value of number 2 is %d\n", num2);
```

```
    return 0;
```

```
}
```

Output:-

Enter the value of number 1: 25

Enter the value of number 2: 15

The value of number 1 is 25

The value of number 2 is 15



Question 03:- Implement algorithm to perform the swapping of two number without using the third variable.

Answer:-

Code:-

```
#include <stdio.h>
int main()
{
    int num1, num2;
    printf("Enter the value of first number: ");
    scanf("%d", &num1);
    printf("Enter the value of second number: ");
    scanf("%d", &num2);
    num1 = num1 + num2;
    num2 = num1 - num2;
    num1 = num1 - num2;
    printf("The value of number 1 is %d\n", num1);
    printf("The value of number 2 is %d\n", num2);
    return 0;
}
```

Q3

Output:-

Enter the value of first number: 35
Enter the value of second number: 15
The value of number 1 is 15
The value of number 2 is 35



Question 04:- Implement algorithm to calculate the factorial of a given number and number is entered by the keyboard.

Answer:-

Code:-

```
#include <stdio.h>
int num, fact = 1;
int factorial(int num) {
    if (num == 1) {
        return 1;
    }
    else {
        fact = num * factorial(num - 1);
        return fact;
    }
}

int main() {
    printf("Enter a number : ");
    scanf("%d", &num);
    printf("Factorial of entered number is %d\n",
    factorial(num));
    return 0;
}
```

Output:-

Enter a number: 5

Factorial of entered number is 120



Question 05:- Implement algorithm to print the Fibonacci series.

Answer:-

Codes-

```
#include <stdio.h>
int fibonacci (int num) {
    if (num == 0) {
        return num;
    }
    else if (num == 1) {
        return num;
    }
    else {
        return fibonacci (num-1) + fibonacci (num-2);
    }
}

int main () {
    int num;
    printf ("Enter a number: ");
    scanf ("%d", &num);
    printf ("Fibonacci series is: ");
    for (int i = 0; i <= num; i++) {
        printf ("%d\t", fibonacci(i));
    }
    return 0;
}
```



Samrat Ashok Technological Institute Varanasi

Name
Sch. No.
Year
Date

outputs-

Enter a number 10

Fibonacci series is: 0 1 1 2 3 5 8 13 21

34 55

SESSIONAL PAPERS



Question 06 :- Implement the algorithm for the traversal of array.

Answer:-

Code-

```
#include <stdio.h>
int main() {
    int num;
    printf("Enter the number of element of array: ");
    scanf("%d", &num);
    int array[num];
    for (int i=0; i<num; i++) {
        printf("Enter the value of %d element: ", i+1);
        scanf("%d", &array[i]);
    }
    for (int i=0; i<num; i++) {
        printf("The value of %d element in array is %d\n",
            i+1, array[i]);
    }
    return 0;
}
```

Output:-

Enter the number of element of array: 5
Enter the value of 1 element: 10
Enter the value of 2 element: 20
Enter the value of 3 element: 30
Enter the value of 4 element: 40



Enter the value of 5 element: 50

The value of 1 element in array is 10

The value of 2 element in array is 20

The value of 3 element in array is 30

The value of 4 element in array is 40

The value of 5 element in array is 50



Question 076 - Implement algorithm to calculate waiting time, Turn around time, average WT, avg TAT of process in FCFS CPU scheduling when arrival time of each process is same i.e. 0.

Answers -

Code

```
#include <stdio.h>
int main () {
    int num;
    printf("Enter the number of process: ");
    scanf("%d", &num);
    int TAT[num], WT[num], BT[num];
    printf("Here we consider the arrival time of each and every process is same i.e. 0 ms \n");
    for (int i=0; i<num; i++) {
        printf("Enter the CPU Burst time of process [%d]: ", i+1);
        scanf("%d", &BT[i]);
    }
    for (int i=0; i<num; i++) {
        printf("CPU Burst time of process [%d] is %d ms \n", i+1, BT[i]);
        WT[i] = 0;
        for (int j=1; j<num; j++) {
            WT[j] = WT[j-1] + BT[j-1];
        }
        int sumWT = 0;
        int sumWT = 0;
        float avgWT = 0;
        for (int i=0; i<num; i++) {
            sumWT = sumWT + WT[i];
        }
    }
}
```



```
printf("Waiting time of process[%d] is %d ms\n", i+1,
WT[i]);
}

avgWT = (float) sumWT/num;
printf("Average Waiting time is %.2f ms\n", avgWT);
TAT[0] = BT[0];
for(int i=1; i<num; i++){
    TAT[i] = TAT[i-1] + BT[i];
    int sumTAT = 0;
    float avgTAT = 0;
    for(int i=0; i<num; i++){
        sumTAT = sumTAT + TAT[i];
        printf("Turn Around time of process[%d] is %d ms\n",
i+1, TAT[i]);
    }
    avgTAT = (float) sumTAT/num;
    printf("Average turn around time is %.2f ms\n", avgTAT);
    return 0;
}
```




Output:-

Enter the number of process: 5

Here we consider the arrival time of each and every process is same i.e. 0 ms

Enter the CPU Burst time of process [1]: 2

Enter the CPU Burst time of process [2]: 6

Enter the CPU Burst time of process [3]: 4

Enter the CPU Burst time of process [4]: 9

Enter the CPU Burst time of process [5]: 12

CPU Burst time of process [1] is 2 ms

CPU Burst time of process [2] is 6 ms

CPU Burst time of process [3] is 4 ms

CPU Burst time of process [4] is 9 ms

CPU Burst time of process [5] is 12 ms

Waiting time of process [1] is 0 ms

Waiting time of process [2] is 2 ms

Waiting time of process [3] is 8 ms

Waiting time of process [4] is 12 ms

Waiting time of process [5] is 21 ms

Average waiting time is 8.60 ms

Turn around time of process [1] is 2 ms

Turn around time of process [2] is 8 ms

Turn around time of process [3] is 12 ms

Turn around time of process [4] is 21 ms

Turn around time of process [5] is 33 ms

Average turn around time is 15.20 ms



Question 08:- Implement algorithm to calculate waiting time, Turn around time of each and every process ~~then~~ in FCFS CPU scheduling when the arrival time of each process is different. Also calculate average waiting time and average turn around time.

Answer:-

Code:-

```
#include <stdio.h>
```

```
int main () {
```

```
    int num;
```

```
    printf("Enter the number of process: ");
```

```
    scanf("%d", &num);
```

```
    int TAT[num], WT[num], BT[num], arrival[num];
```

```
    for(int i=0; i<num; i++) {
```

```
        printf("Enter the arrival time of process [%d]: ", i+1);
```

```
        scanf("%d", &arrival[i]);
```

```
    }
```

```
    for(int i=0; i<num; i++) {
```

```
        printf("Enter the CPU burst time of process [%d]: ", i+1);
```

```
        scanf("%d", &BT[i]);
```

```
    }
```

```
    for(int i=0; i<num; i++) {
```

```
        printf("Arrival time of process [%d] is %d ms in ", i+1, arrival[i]);
```

```
    }
```

```
    printf("CPU burst time of process [%d] is %d ms in ", i+1, BT[i]);
```

```
    WT[0] = 0;
```

```
    for(int i=1; i<num; i++)
```

```
        WT[i] = WT[i-1] + BT[i-1] - arrival[i] + arrival[i-1];
```




```
int sumWT = 0;
float avgWT = 0;
for (int i = 0; i < num; i++) {
    sumWT = sumWT + WT[i];
    printf("Waiting time of process [%d] is %d ms\n", i+1,
    WT[i]);
}
avgWT = (float) sumWT / num;
printf("Average waiting time is %.2f ms\n", avgWT);
TAT[0] = BT[0];
for (int i = 1; i < num; i++)
    TAT[i] = TAT[i-1] + BT[i] - arrival[i] + arrival[i-1];
int sumTAT = 0;
float avgTAT = 0;
for (int i = 0; i < num; i++) {
    sumTAT = sumTAT + TAT[i];
    printf("Turn around time of process [%d] is %d ms\n",
    i+1, TAT[i]);
}
avgTAT = (float) sumTAT / num;
printf("Average turn around time is %.2f ms\n", avgTAT);
return 0;
```

3



Output:-

Enter the number of process: 5

Enter the Arrival time of process [1]: 0

Enter the Arrival time of process [2]: 1

Enter the Arrival time of process [3]: 2

Enter the Arrival time of process [4]: 3

Enter the Arrival time of process [5]: 6

Enter the cpu burst time of process [1]: 2

Enter the cpu burst time of process [2]: 6

Enter the cpu burst time of process [3]: 4

Enter the cpu burst time of process [4]: 9

Enter the cpu burst time of process [5]: 12

Arrival time of process [1] is 0 ms

Arrival time of process [2] is 1 ms

Arrival time of process [3] is 2 ms

Arrival time of process [4] is 3 ms

Arrival time of process [5] is 6 ms

CPU burst time of process [1] is 2 ms

CPU burst time of process [2] is 6 ms

CPU burst time of process [3] is 4 ms

CPU burst time of process [4] is 9 ms

CPU burst time of process [5] is 12 ms

Waiting time of process [1] is 0 ms

Waiting time of process [2] is 1 ms

Waiting time of process [3] is 6 ms

Waiting time of process [4] is 9 ms

Waiting time of process [5] is 15 ms

Average waiting time is 6.20 ms.

SESSIONAL PAPERS

Name Opthank Agrawal

Sch. No.

Year

Date

Turn around time of process [1] is 2 ms

Turn around time of ~~process~~ process [2] is 7 ms

Turn around time of process [3] is 10 ms

Turn around time of process [4] is 18 ms

Turn around time of ~~from~~ online process [5] is 27 ms

Average turn ~~arounding~~ around time is 12.80 ms