Bahria University,

Karachi Campus



LAB EXPERIMENT NO.

\_\_\_08\_\_\_

LIST OF TASKS

|  |  |
| --- | --- |
| TASK NO | OBJECTIVE |
| **01** | Explore HTOP, including its options. Attach outputs for the same. |
| **02** | Write a multithreaded C program for performing summation of numbers. |
| 03 | Write a program which make 4 threads. Each thread will print one table out of [5678] up to 1000. |

Submitted On

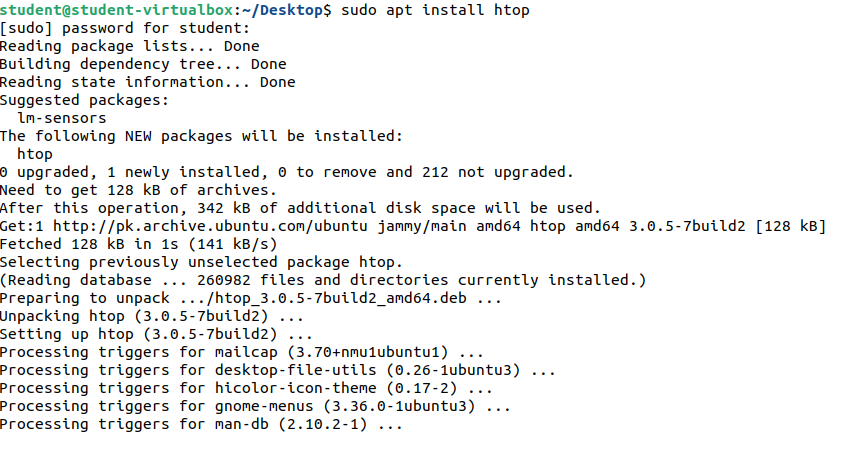
11-05-2023

(Date: DD/MM/YY)

**Task 01:** Explore HTOP, including its options. Attach outputs for the same.

**Source Code:**

**Solution:**

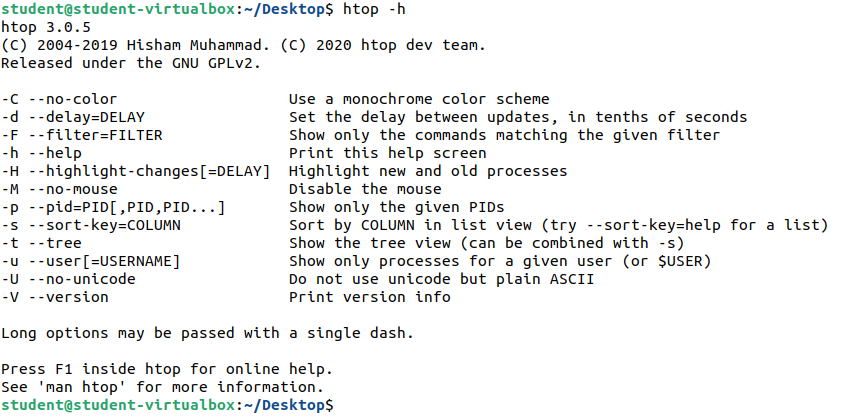
HTOP command in Linux System is a command line utility that allows the user to interactively monitor the system’s vital resources or server’s processes in real time.

Table

Description automatically generated

**Output:**

* Htop -h ( Help )



* F5 Tree

Table

Description automatically generated

**Task 02:** Write a multithreaded C program for performing summation of numbers.

**Solution:**

#include <stdio.h>

#include <unistd.h>

#include <pthread.h>

int sum;

int array[2];

void \*AddNumbers(void \*arg);

void main(){

pthread\_t thread1;

printf("Enter Number 1: \n");

scanf("%d", &array[0]);

printf("Enter Numnber 2: \n");

scanf("%d", &array[1]);

pthread\_create(&thread1, NULL, AddNumbers, array);

pthread\_join(thread1, NULL);}

void \*AddNumbers (void \*arg){

int \* arr = (int \*) arg;

int n1= arr[0];

int n2= arr[1];

sum = n1 + n2;

printf("\nSum of both numbers = %d \n", sum);

pthread\_exit(0);}

**Output:**

**Text

Description automatically generated**

**Task 03:** Write a program which make 4 threads. Each thread will print one table out of [5678] up to 1000.

**Source Code:**

#include <stdio.h>

#include <unistd.h>

#include <pthread.h>

int num = 5;

int array[3];

void \* Tables(void \*arg);

void main(){

pthread\_t thread1;

pthread\_t thread2;

pthread\_t thread3;

pthread\_t thread4;

for (int i = 0; i < 4; i++){

array[i] = num;

num++;}

num = 5;

pthread\_create(&thread1, NULL, Tables, array);

pthread\_join(thread1, NULL);

num++;

pthread\_create(&thread2, NULL, Tables, array);

pthread\_join(thread2, NULL);

num++;

pthread\_create(&thread3, NULL, Tables, array);

pthread\_join(thread3, NULL);

num++;

pthread\_create(&thread4, NULL, Tables, array);

pthread\_join(thread4, NULL);}

void \*Tables(void \*arg){

for (int i = 1; i <= 1000; i++){

switch (num){

case 5:

printf("%d x %d = %d\n", array[0], i, array[0] \* i);

break;

case 6:

printf("%d x %d = %d\n", array[1], i, array[1] \* i);

break;

case 7:

printf("%d x %d = %d\n", array[2], i, array[2] \* i);

break;

case 8:

printf("%d x %d = %d\n", array[3], i, array[3] \* i);

break;}}

pthread\_exit(0);}

**Output:**

**A picture containing background pattern

Description automatically generatedA picture containing background pattern

Description automatically generated**

**A picture containing text

Description automatically generatedA picture containing text

Description automatically generated**