ROHIT RAJ

1RV17CS125

PADP LAB, C2

<div align="center">

PROGRAM-4

</div>

CODE:

```c
#include <stdio.h>

#include <error.h>

#include <gd.h>

#include<string.h>

#include<omp.h>


int main(int argc, char **argv)
{
        FILE *fp,*fp1 = {0};

        gdImagePtr img;

        char iname[15];

        char oname[15];

        int color, x, y, i=0;

        int red, green, blue,tmp,tid;

        long w,h;


        color = x = y = w = h = 0;

        red = green = blue = 0;


        //char inputnames[4][15] =
{"input1.png","input2.png","input3.png","input4.png"};
        //char outputnames[4][15] =
{"output1.png","output2.png","output3.png","output4.png"};
        omp_sched_t def_sched; int def_chunk_size;
```

```c
omp_get_schedule(&def_sched,&def_chunk_size);
printf("Deafault %d %d \n",def_sched,def_chunk_size);
printf("Size\t\tDefault\t\tStatic\t\tDynamic\t\tGuided\n");
for(int i=0;i<4;i++){
        sprintf(iname,"in%d.png",i+1);

        for(int sched=0x0; sched<=0x3; sched++){
                fp = fopen(iname,"r");
                sprintf(oname,"Output%d%d.png",i+1,sched);
                img = gdImageCreateFromPng(fp);
                w = gdImageSX(img);
                h = gdImageSY(img);

                if(sched == 0x0){
                        printf("%ldx%ld\t",w,h);
                        if(i<=1) printf("\t");
                        omp_set_schedule(def_sched, def_chunk_size);
                }
                else
                        omp_set_schedule(sched, 0);

                double t = omp_get_wtime();
                #pragma omp parallel for private(y,color,red,green,blue,tmp,tid)
//schedule(dynamic)
                for(x = 0; x < w; x++)
                {

                        for(y = 0; y < h; y++)
                        {
                                tid=omp_get_thread_num();
```

```c
                    color=gdImageGetPixel(img, x, y);

                    red   = gdImageRed(img, color);

                    green = gdImageGreen(img, color);

                    blue  = gdImageBlue(img, color);

                    tmp = (red+green+blue)/ 3;

                    red = green = blue = tmp;

                    color = gdImageColorAllocate(img, red, green, blue);

                    gdImageSetPixel(img, x, y, color);

                t = omp_get_wtime() - t;

                //printf("Output name %s",oname);

                fp1=fopen(oname,"w");

                gdImagePng(img, fp1);

                fclose(fp1);


                gdImageDestroy(img);

                printf("%.6f\t",t);

            }

            printf("\n");

        }

        return 0;

    }
```

OUTPUT:


```
rohit@Rohit: /mnt/c/users/rohit/Desktop/New folder
rohit@Rohit:/mnt/c/users/rohit/Desktop/New folder$ gcc prog4.c -fopenmp -lgd
rohit@Rohit:/mnt/c/users/rohit/Desktop/New folder$ ./a.out
Deafault 2 1
Size            Default         Static          Dynamic         Guided
800x500         0.011171        0.008589        0.008697        0.008921
912x513         0.010275        0.023642        0.010733        0.027787
1920x1080       0.068456        0.059489        0.057786        0.058231
1800x1350       0.070451        0.076877        0.070236        0.071564
rohit@Rohit:/mnt/c/users/rohit/Desktop/New folder$
```