**Student Name: Divyanshu Tiwari**

**Student ID : 11802090**

**Email Address: sunnyt0225@gmail.com**

**GitHub Link: https://github.com/deadshot674gam/CSE316.git**

**Problem-**Ten students (a, b, c, d, e, f, g, h, i, j) are going to attend an event. There are lots of gift shops, they all are going to the gift shops and randomly picking the gifts. After picking the gifts they are randomly arriving in the billing counter. The accountant gives the preference to that student who has maximum number of gifts.

Solution Code:

#include<stdio.h>

#include<conio.h>

#include <stdlib.h>

#include <unistd.h> //Header file for sleep().

#include <pthread.h>

typedef struct Queue{

int item;

char name;

}Queue;

void \*myThreadFun(void \*vargp) {

system("CLS");

printf(" \n");

printf(" \n");

printf("\t\t\t\t ================================================= \n");

sleep(1);

printf(" \t\t\t\t||\t \_\_\_(Billing System)\_\_\_\t ||\n");

printf("\t\t\t\t ================================================= \n");

sleep(1);

system("COLOR 17");

sleep(1);

system("COLOR 47");

sleep(1);

system("COLOR 9");

printf("\n\n\n\n\n\n\n\n\t\t\t (Press any key to continue)\n");

printf(" ---------------------------------------------------------------------------------------------------------------------");

getch();

return NULL;

}

void bubbleSort(Queue\* a,int size){//O(n2)

int i,j;

for(i=0;i<size;i++){

for(j=0;j<size-i-1;j++){

if(a[j].item<a[j+1].item){

Queue temp=a[j+1];

a[j+1]=a[j];

a[j]=temp;

}

}

}

}

void main(){

pthread\_t thread\_id;

pthread\_create(&thread\_id, NULL, myThreadFun, NULL);

pthread\_join(thread\_id, NULL);

system("CLS");

Queue Q[10];

int a[10],sum=0;

printf("Enter the number of item all ten kids (with names a,b,c,d,e,f,g,h,i,j) have one by one:\n");

for(int i=0;i<10;i++){//O(n)

scanf("%d %c",&Q[i].item,&Q[i].name);

}

for(int i=0;i<10;i++){//O(n)

sum=sum+Q[i].item;

}

bubbleSort(Q,10);

for(int i=0;i<10;i++){//O(n)

printf("%c ",Q[i].name);

}

printf("\nTotal Time taken by Billing person: %d",sum);

}

**Problem in terms of Operating System**

This problem is an example of Non - pre-emptive Scheduling as Billing person will only bill one person at a time and he can be disrupted by other person.

**Algorithm Used by me**

I used basic Algorithm to perform this task :

* Bubble Sort for sorting the students in descending order according to number of gifts they picked.
* Queue named data structure for storing both no of items and name of student.

**Description**

Only ten students are there, they are being prioritized based on number of gifts they are carrying, the sequence will be decided in descending of number of gifts each of them is carrying.

**Additional Algorithm**

typedef struct Queue { // Data Structure Used

int item;

char name;

} Queue;

void bubbleSort(Queue\* a,int size){ //Sorting algorithm

int i,j;

for(i=0;i<size;i++){

for(j=0;j<size-i-1;j++){

if(a[j].item<a[j+1].item){

Queue temp=a[j+1];

a[j+1]=a[j];

a[j]=temp;

}

}

}

}

**Boundary Conditions**

Here array of Data Structure Queue is of fix size in case of changing size manual code should be terminated. I know how to allot that Dynamically but as the question says 10 students that is specified.

It only gives the execution order and total time taken.

**Test case**

Enter the number of item all ten kids (with names a, b, c, d, e, f, g, h, i, j) have one by one:

10 a

20 b

30 c

40 d

15 e

7 f

2 g

19 h

100 i

1 j

**Output**

i d c b h e a f g j

Total Time taken by Billing person: 244