CSE-316

OPERATING SYSTEM PROJECT REPORT

QUE NO-18

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. Create a C or Java program to define order of billed students?

Submitted by:

SHAIK ABDUL KHADAR JILANI

Reg No:11815233

Roll No:19

Section: K18EN

Email: jilanishaik2804@gmail.com

Github: https://github.com/jilani-shaik/simulated-assignment

CODE

```
#include<stdio.h>

int main()
```

```
{
      int a[10],i,h,k;
      char b[10],k1;
     for(i=0;i<10;i++)
      b[i]=(char)(97+i);
      printf("Enter Gifts taken by 10 students\n");
      for(i=0;i<10;i++)
{
        printf("for %c student : \t",b[i]);
      scanf("%d",&a[i]);
}
h=0;
for(;h==0;)
{
   h=1;
  for(i=0;i<9;i++)
{
 if(a[i]<a[i+1])
 {
  k=a[i];
  a[i]=a[i+1];
  a[i+1]=k;
  k1=b[i];
  b[i]=b[i+1];
  b[i+1]=k1;
  h=0;}
  }
  }
printf("The order is \n");
printf("-----------------------------------------------\n");
printf("Students \t   no of gifts\n");
printf("-----------------------------------------------\n");
for(i=0;i<10;i++)
{


printf(" %c \t\t\t %d\n", b[i],a[i]);
}
}
```

1.Description

Non-Preemptive Scheduling: **Non-preemptive Scheduling** is used when a process terminates, or a process switches from running to waiting state. In this **scheduling**, once the resources (CPU cycles) is allocated to a process, the process holds the CPU till it gets terminated or it reaches a waiting state.

Priority scheduling algorithm is a popular scheduling technique for job assignments. In this the process having higher priority is executed first. The higher priority being the no closer to 0. If any of the process having same priority then they are assigned based on their arrival time. Priority can be decided based on memory requirements, time requirements or any other resource requirement.

Advantages of Priority Scheduling

1. The priority of process is selected on the basis of memory requirement, user preference or the requirement of time.
2. Processes are executed on the basis of priority. So high priority does not need to wait for long which saves time.
3. It is easy to use.
4. It is a user-friendly algorithm.
5. Simple to understand.
6. It has reasonable support for priority.

2.Algorithm

```
Step 1      for i ← 1 to n-1 do {
Step 2          for j ← 1 to n-i do {
Step 3          if (A[j+1] < A[j]) swap A[j] and A[j+1] ;
            }
        }
Step 4    exit
```

3.COMPLEXITY

Complexity of input and output: O(n)

Inside loops

Complexity of Bubble sorting: O(n^2)

4.CONSTRINTS TAKEN

i, j-loop

5.ADDITIONAL ALGORITHM

Apart from the main algorithm no additional algorithm is used.

6.Boundary Condition

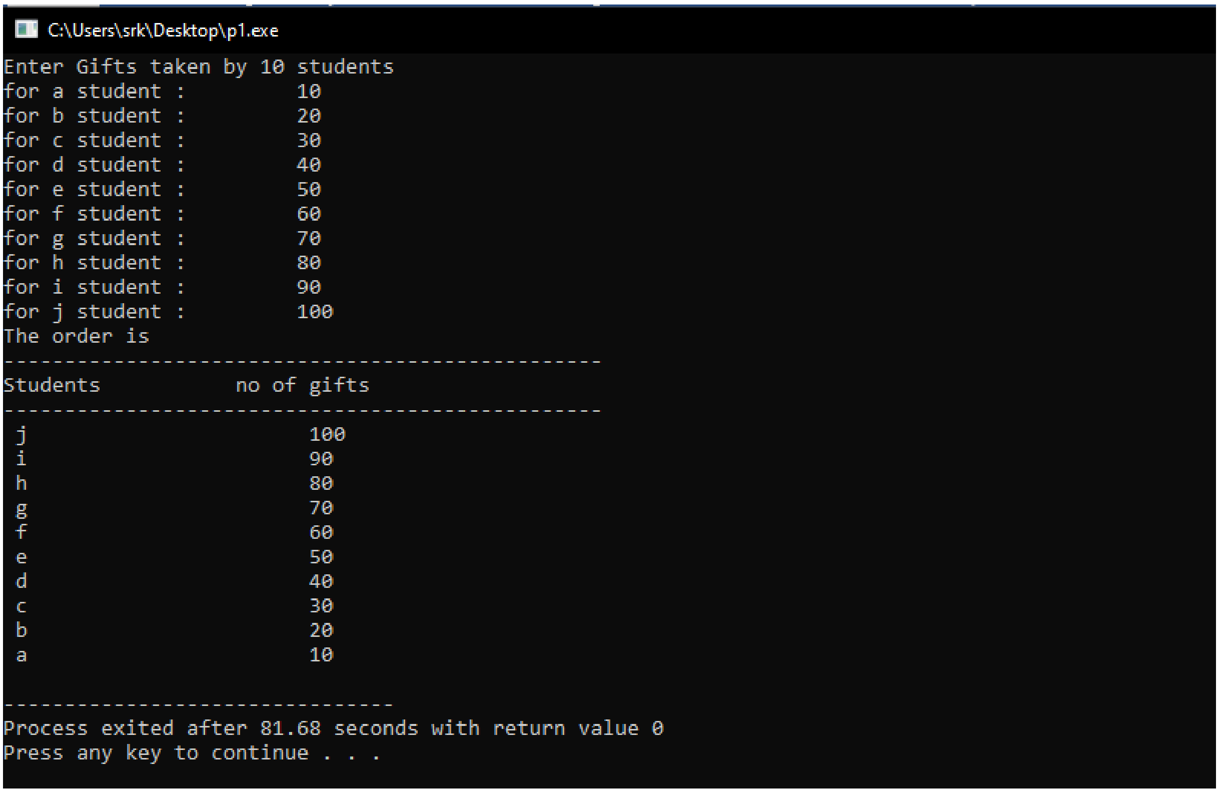
1. The maximum array size is given as 10 so values greater than that cannot be processed.
2. Likewise, for the priority also the maximum priority limit for the process should be less than 10.
3. Detection of negative values is not enable so put all values above 0.

7.TEST CASES AND OUTPUT

Test Case 1:when students pick gifts randomly

Accountant prefers the individual who has the maximum number of gifts for billing

.

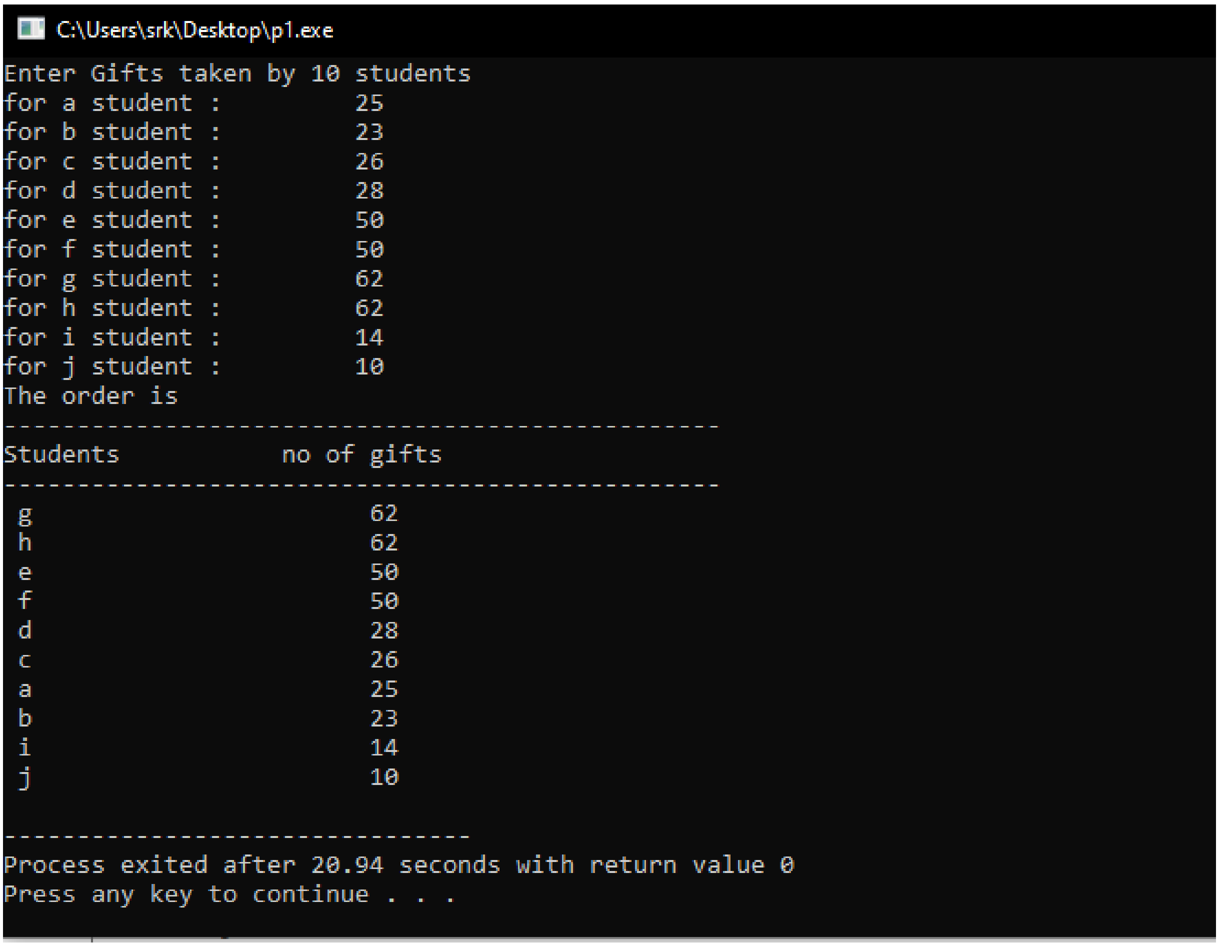


```
Enter Gifts taken by 10 students
for a student :         10
for b student :         20
for c student :         30
for d student :         40
for e student :         50
for f student :         60
for g student :         70
for h student :         80
for i student :         90
for j student :         100
The order is
-------------------------------------------------
Students           no of gifts
-------------------------------------------------
 j                       100
 i                       90
 h                       80
 g                       70
 f                       60
 e                       50
 d                       40
 c                       30
 b                       20
 a                       10

--------------------------------
Process exited after 81.68 seconds with return value 0
Press any key to continue . . .
```

Test Case 2: When two or more students pick same number of gifts then

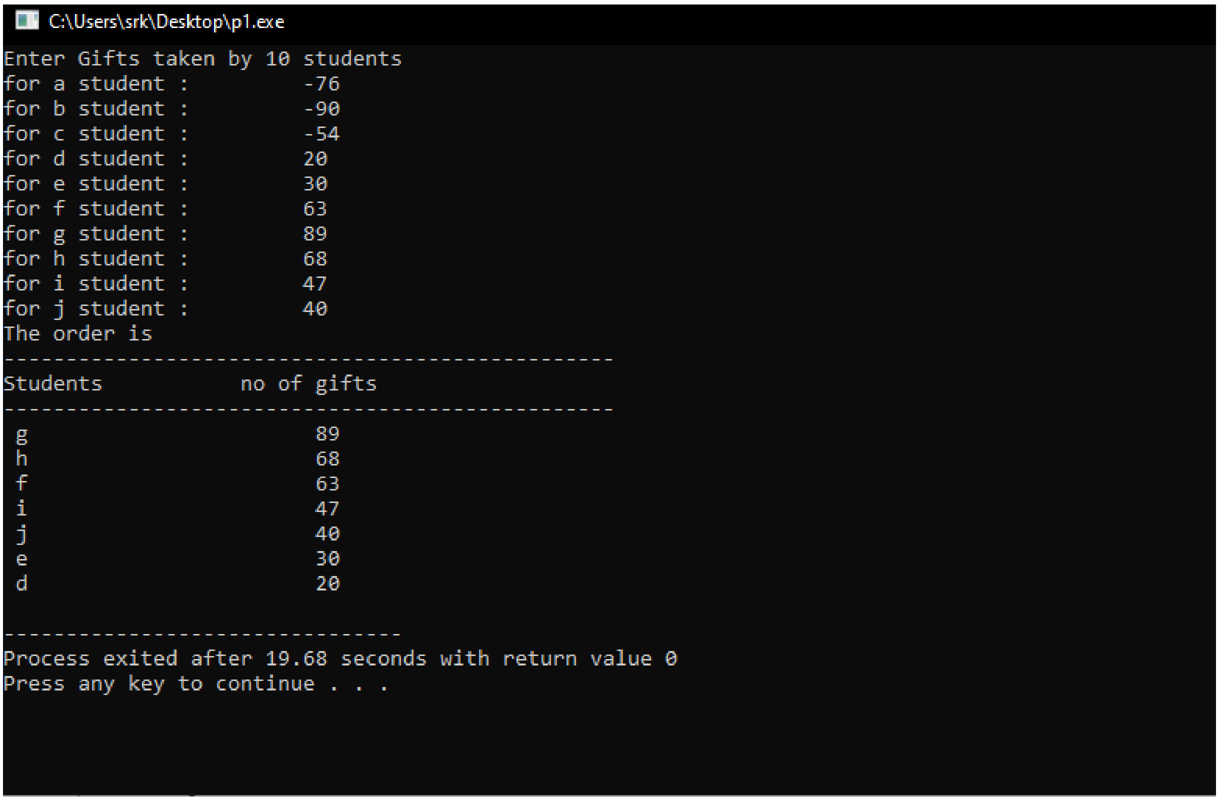
Account prefer the students with maximum number of gifts and then remaining students .



```
Enter Gifts taken by 10 students
for a student :         25
for b student :         23
for c student :         26
for d student :         28
for e student :         50
for f student :         50
for g student :         62
for h student :         62
for i student :         14
for j student :         10
The order is
-------------------------------------------------
Students           no of gifts
-------------------------------------------------
 g                       62
 h                       62
 e                       50
 f                       50
 d                       28
 c                       26
 a                       25
 b                       23
 i                       14
 j                       10

--------------------------------
Process exited after 20.94 seconds with return value 0
Press any key to continue . . .
```

Test Case 3: When user enters negative values accountant will take only positive values.



```
Enter Gifts taken by 10 students
for a student :         -76
for b student :         -90
for c student :         -54
for d student :         20
for e student :         30
for f student :         63
for g student :         89
for h student :         68
for i student :         47
for j student :         40
The order is
-------------------------------------------------
Students           no of gifts
-------------------------------------------------
 g                       89
 h                       68
 f                       63
 i                       47
 j                       40
 e                       30
 d                       20

--------------------------------
Process exited after 19.68 seconds with return value 0
Press any key to continue . . .
```