Innovative Assignment

Name and Roll No. : Aditya Singh 20BCE007

Course Code and Name : 2CS403 Operating Systems

AIM:

Python Code for Memory Management(best fit, worst fit, next fit and first fit)

CODE:

First Fit:

```
suit=[]
able=[]
def firstfit(bf, pf):
    pno=len(pf)
    bno=len(bf)
    a = [-1] * pno
    for i in range(0,pno):
        for j in range(0,bno):
            if bf[j] >= pf[i]:
                bf[j]-=pf[i]
                a[i]=j
                break
        if a[i] != -1:
            for k in range(0,a[i]):
                print("  \t  \t  ",end="")
            print("P{}: {}".format((i + 1),(pf[i])))

        else:
            print("P{}: {}       Not Allocated".format((i + 1),(pf[i])))
        print("\nHole: ",bf)
        print("\n")
    count=0
    for i in range(0,pno):
        if a[i]==-1:
            count+=1
```

```
    if count>0:
        suit.append("F")
        able.append("F")
    else:
        suit.append(max(bf))
        able.append(min(bf))
```

NEXT FIT:

```
def nextfit(bn, pn):
    m=len(bn)
    n=len(pn)
    a = [-1] * n
    j = 0
    for i in range(n):
        while j < m:
            if bn[j] >= pn[i]:
                a[i] = j
                bn[j] -= pn[i]
                break
            j = (j + 1) % m
    for i in range(n):
        if a[i] != -1:
            for k in range(0,a[i]):
                print("  \t  \t  ",end="")
            print("P{}: {}".format((i + 1),(pn[i])))

        else:
            print("P{}: {}       Not Allocated".format((i + 1),(pn[i])))
        print("\nHole: ",bn)
        print("\n")
    count=0
    for i in range(0,n):
        if a[i]==-1:
```

```
            count+=1


    if count>0:
        suit.append("F")
        able.append("F")
    else:
        suit.append(max(bn))
        able.append(min(bn))
```

BEST FIT:

```
def bestfit(bb, pb):
    pno=len(pb)
    bno=len(bb)
    a = [-1] * pno
    for i in range(0,pno):
        c=[]
        for j in range(0,bno):
            if bb[j] >= pb[i]:
                c.append(bb[j])
        for j in range(0,bno):
            if bb[j] >= pb[i] and bb[j]==min(c):
                bb[j]-=pb[i]
                a[i]=j
                break
        if a[i] != -1:
            for k in range(0,a[i]):
                print("  \t  \t  ",end="")
            print("P{}: {}".format((i + 1),(pb[i])))


        else:
            print("P{}: {}       Not Allocated".format((i + 1),(pb[i])))
        print("\nHole: ",bb)
        print("\n")
    count=0
```

```
    for i in range(0,pno):
        if a[i]==-1:
            count+=1

    if count>0:
        suit.append("F")
        able.append("F")
    else:
        suit.append(max(bb))
        able.append(min(bb))
```

WORST FIT:

```
def worstfit(bw, pw):
    pno=len(pw)
    bno=len(bw)
    b_=bw.copy()
    a = [-1] * pno
    for i in range(0,pno):
        for j in range(0,bno):
            if b_[j] >= p[i] and b_[j]==max(b_):
                b_[j]-=p[i]
                a[i]=j
                break
        if a[i] != -1:
            for k in range(0,a[i]):
                print("  \t  \t  ",end="")
            print("P{}: {}".format((i + 1),(p[i])))

        else:
            print("P{}: {}       Not Allocated".format((i + 1),(p[i])))
        print("\nHole: ",b_)
        print("\n")
    count=0
    for i in range(0,pno):
```

```
        if a[i]==-1:
            count+=1

    if count>0:
        suit.append("F")
        able.append("F")
    else:
        suit.append(max(bb))
        able.append(min(bb))
```

SUITABLE:

```
def suitable():
    fits=["First Fit","Next Fit","Best Fit","Worst Fit"]
    print("\nMost suitable: ",end="")
    c=[]
    s=[]
    for i in range(0,len(fits)):
        if suit[i] != 'F' and able[i]!= 'F':
            c.append(suit[i])
            s.append(able[i])

    for i in range(0,len(fits)):
        if suit[i] == max(c) and able[i]==min(s):
            print(fits[i],end=" ")
    print("\n")
```

MAIN:

```
if __name__ == '__main__':
    b=[]
    p=[]
    n=int(input("Enter the number of processes: "))
    m=int(input("Enter the number of holes: "))
    print("Enter size of processes:")
    for i in range(0,n):
```

```
        print("P{} = ".format(i+1),end="")
        a=int(input())
        p.append(a)
    print("Enter size of hole:")
    for i in range(0,m):
        print("Hole {} = ".format(i+1),end="")
        a=int(input())
        b.append(a)
    bf=b.copy()
    bw=b.copy()
    bn=b.copy()
    bb=b.copy()
    n=1
    while n>0:
        print("1. First Fit\n2. Next Fit\n3. Best Fit\n4. Worst Fit\n5. Most
Suitable\n6. Exit")
        fit=["First Fit","Next Fit","Best Fit","Worst Fit"]
        ch=int(input("Enter choice: "))
        if ch==1:
            print("\n==========================")
            print("========FIRST FIT=========")
            print("==========================")
            print("\n\n")
            for i in range(0,m):
                print("||\t{}\t".format(bf[i]),end="")
            print("||")
            firstfit(bf,p)
        if ch==2:
            print("\n==========================")
            print("=========NEXT FIT=========")
            print("==========================")
            print("\n\n")
            for i in range(0,m):
                print("||\t{}\t".format(bn[i]),end="")
```

```
    print("||")
    nextfit(bn,p)
if ch==3:
    print("\n==========================")
    print("========BEST FIT=========")
    print("==========================")
    print("\n\n")
    for i in range(0,m):
        print("||\t{}\t".format(bb[i]),end="")
    print("||")
    bestfit(bb,p)
if ch==4:
    print("\n==========================")
    print("=========WORST FIT=========")
    print("==========================")
    print("\n\n")
    for i in range(0,m):
        print("||\t{}\t".format(bw[i]),end="")
    print("||")
    worstfit(bw,p)
if ch==5:
    suitable()
if ch==6:
    print("\nThank You!!\n20BCE007 Aditya")
    n=0
```

OUTPUT:

INPUT VALUES:

```
Enter the number of processes: 4
Enter the number of holes: 5
Enter size of processes:
P1 = 118
P2 = 350
P3 = 156
P4 = 499
Enter size of hole:
Hole 1 = 100
Hole 2 = 600
Hole 3 = 200
Hole 4 = 300
Hole 5 = 700
1. First Fit
2. Next Fit
3. Best Fit
4. Worst Fit
5. Most Suitable
6. Exit
```

FIRST FIT:

```
||      100     ||      600     ||      200     ||      300     ||      700     ||
                  P1: 118

Hole:  [100, 482, 200, 300, 700]


                  P2: 350

Hole:  [100, 132, 200, 300, 700]


                                  P3: 156

Hole:  [100, 132, 44, 300, 700]


                                                                  P4: 499

Hole:  [100, 132, 44, 300, 201]
```

NEXT FIT:

```
||      100     ||      600     ||      200     ||      300     ||      700     ||
                  P1: 118

Hole:  [100, 132, 44, 300, 201]


                  P2: 350

Hole:  [100, 132, 44, 300, 201]


                               P3: 156

Hole:  [100, 132, 44, 300, 201]


                                                          P4: 499

Hole:  [100, 132, 44, 300, 201]
```

BEST FIT:

```
||      100     ||      600     ||      200     ||      300     ||      700     ||
                                P1: 118

Hole:  [100, 600, 82, 300, 700]

                P2: 350

Hole:  [100, 250, 82, 300, 700]

                P3: 156

Hole:  [100, 94, 82, 300, 700]

                                                                P4: 499

Hole:  [100, 94, 82, 300, 201]
```

WORST FIT:

```
||      100     ||      600     ||      200     ||      300     ||      700     ||
                                                                P1: 118

Hole:  [100, 600, 200, 300, 582]

                P2: 350

Hole:  [100, 250, 200, 300, 582]

                                                                P3: 156

Hole:  [100, 250, 200, 300, 426]

P4: 499       Not Allocated

Hole:  [100, 250, 200, 300, 426]
```

MOST SUITABLE:

```
1. First Fit
2. Next Fit
3. Best Fit
4. Worst Fit
5. Most Suitable
6. Exit
Enter choice: 5

Most suitable: First Fit Next Fit
```