**Practical : 9**

**Aim: Implementation of First-Fit Memory Allocation Algorithm**

**Program:**

**#include <stdio.h>**

**struct process**

**{**

**int size; int flag; int holeid;**

**} p[10];**

**struct hole**

**{**

**int size;**

**int actual;**

**} h[10];**

**int main()**

**{**

**int i, np, nh, j, sum=0;**

**char ans='y';**

**//do{**

**label:**

**printf("Enter the number of Holes : ");**

**scanf("%d", &nh);**

**for(i=0; i<nh; i++)**

**{**

**printf("Enter size for hole H%d : ",i);**

**scanf("%d", &h[i].size);**

**h[i].actual = h[i].size;**

**}**

**printf("\nEnter number of process : " );**

**scanf("%d",&np);**

**for(i=0;i<np;i++)**

**{**

**printf("enter the size of process P%d : ",i);**

**scanf("%d", &p[i].size);**

**p[i].flag = 0;**

**}**

**for(i=0; i<np; i++)**

**{**

**for(j=0; j<nh; j++)**

**{**

**if(p[i].flag != 1)**

**{**

**if(p[i].size <= h[j].size)**

**{**

**p[i].flag = 1; p[i].holeid = j; h[j].size -= p[i].size;**

**}**

**}**

**}**

**}**

**printf("\n\tFirst fit\n");**

**for(i=0;i<70;i++)**

**{**

**printf("-");**

**}**

**printf("\n|Process | PSize |\t Hole \t| Actual\t| Available |\n");**

**for(i=0;i<70;i++)**

**{**

**printf("-");**

**}**

**for(i=0; i<np; i++)**

**{**

**if(p[i].flag != 1)**

**{**

**printf("\n| P%d |\t%d |\t Not allocated |\t -\t|\t-\t|", i, p[i].size);**

**printf("\n---------------------------------------------------------------------");**

**h[p[i].holeid].size=0;**

**}**

**else**

**{**

**printf("\n| P%d |\t%d |\t H%d \t|\t%d\t| \t %d |", i, p[i].size, p[i].holeid,h[p[i].holeid].actual,h[p[i].holeid].size);**

**sum = sum + h[p[i].holeid].size;**

**printf("\n---------------------------------------------------------------------");**

**}**

**}**

**printf("\n");**

**printf("Total available size is : %d \n",sum);**

**printf( " \nDo you want to continue (Y/N)?\n");**

**printf( "You must type a 'Y' or an 'N':");**

**scanf(" %c",&ans);**

**if( ans=='y' || ans=='Y')**

**{**

**goto label;**

**}**

**else**

**{**

**return 0;**

**}**

**}**

**Output:**

