ABHINAV U

S5 CSE

Roll no:05

```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
struct node
{
char name[20];
char type;
struct node *next[10]; //Multiple pointer for storing more than one file or directory in same directory
struct node *prev;
};
struct node *head = NULL; //Pointer containing address of first node in linked list
struct node *current = NULL; //Pointer for traversing the linked list
struct node *prev = NULL;
struct node *search = NULL; //Pointer for searching duplication of file or directory name
//Function to perform single level file operation
void single_level()
{
char file_name[20],c;
int i,k;
printf("\n*****Single Level*****");
struct node *new = (struct node*) malloc(sizeof(struct node));
strcpy(new->name,"root");
new->type = 'd';
new->prev = NULL;
for(i=0;i<10;i++)
{
new->next[i] = NULL;
}
```

```
head = new;
current = head;
k = 0;
do
{
printf("\nEnter file name: ");
scanf("%s",file_name);
search = head;
i = 0;
while(i<10 && search->next[i] != NULL)
{
if(strcmp(search->next[i]->name,file_name) == 0)
{
printf("\nFile Name already exist!");
goto skip0;
}
i++;
}
struct node *new = (struct node*) malloc(sizeof(struct node));
strcpy(new->name,file_name);
new->type = 'f';
new->prev = head;
head->next[k++] = new;
for(i=0;i<10;i++)
{
new->next[i] = NULL;
}
skip0:
printf("\nDo you want to continue creating files (Y/N)? ");
scanf(" %c",&c);
}
```

```
while(c == 'Y' || c =='y');
printf("\nThe files entered are:");
search = head;
i = 0;
while(i<10 && search->next[i] != NULL)
{
printf("\n%s",search->next[i]->name);
i++;
}
}
void two_level()
{
char string[20],c0,c1;
int i,j,k;
printf("\n*****Two Level*****");
struct node *new = (struct node*) malloc(sizeof(struct node));
strcpy(new->name,"root");
new->type = 'd';
new->prev = NULL;
for(i=0;i<10;i++)
{
new->next[i] = NULL;
}
head = new;
current = head;
j = 0;
do
{
printf("\nEnter name of user: ");
scanf("%s",string);
search = head;
```

```
i = 0;
while(i<10 && search->next[i] != NULL)
{
if(strcmp(search->next[i]->name,string) == 0)
{
printf("\nUser Name already exist!");
goto skip1;
}
i++;
}
struct node *new = (struct node*) malloc(sizeof(struct node));
strcpy(new->name,string);
new->type = 'd';
new->prev = head;
head->next[j++] = new;
for(i=0;i<10;i++)
{
new->next[i] = NULL;
}
current = new;
k = 0;
do
{
printf("\nEnter file name for user %s: ",current->name);
scanf("%s",string);
search = current;
i = 0;
while(i<10 && search->next[i] != NULL)
{
if(strcmp(search->next[i]->name,string) == 0)
{
```

```
printf("\nFile Name already exist!");
goto skip2;
}
i++;
}
struct node *new = (struct node*) malloc(sizeof(struct
node));
strcpy(new->name,string);
new->type = 'f';
new->prev = current;
current->next[k++] = new;
for(i=0;i<10;i++)
{
new->next[i] = NULL;
}
skip2:
printf("\nDo you want to continue creating files for user %s (Y/N)? ",current->name);
scanf(" %c",&c0);
}
while(c0 == 'Y' || c0 =='y');
current = head;
skip1:
printf("\nDo you want to continue creating users (Y/N)? ");
scanf(" %c",&c1);
}
while(c1 == 'Y' || c1 =='y');
printf("\nUsers entered are:");
search = head;
i = 0;
while(i<10 && search->next[i] != NULL)
{
```

```
printf("\n%s",search->next[i]->name);
i++;
}
printf("\n");
search = head;
i = 0;
while(i<10 && search->next[i] != NULL)
{
printf("\nFiles entered in %s:",search->next[i]->name);
j = 0;
while(j<10 && search->next[i]->next[j] != NULL)
{
printf("\n%s",search->next[i]->next[j]->name);
j++;
}
printf("\n");
i++;
}
}
void t_user(struct node *head)
{
int i;
char string[20];
current = head;
printf("\nEnter name of User: ");
scanf("%s",string);
i = 0;
search = current;
while(i<10 && search->next[i] != NULL)
{
if(strcmp(search->next[i]->name,string) == 0)
```

```
{
printf("\nUser name already used!");
goto skip3;
}
i++;
}
struct node *new = (struct node*) malloc(sizeof(struct node));
strcpy(new->name,string);
new->type = 'd';
new->prev = head;
for(i=0;i<10;i++)
{
new->next[i] = NULL;
}
search = current;
for(i=0;i<10;i++)
{
if(search->next[i] == NULL)
{
search->next[i] = new;
break;
}
}
if(i == 10)
{
printf("\nRoot directory is full!");
}
skip3:
printf("\n");
}
void t_directory(struct node *current)
```

```
{
int i,ch;
char string[20];
printf("\nWhere do you want to create a directory?");
if(current != head)
{
printf("\n-1.Here");
}
i = 0;
search = current;
while(i<10 && search->next[i] != NULL)
{
if(search->next[i]->type != 'f')
{
printf("\n%d.%s",i,search->next[i]->name);
}
i++;
}
printf("\n");
scanf("%d",&ch);
if(ch == -1)
{
printf("\nEnter name of directory: ");
scanf("%s",string);
search = current;
i = 0;
while(i<10 && search->next[i] != NULL)
{
if((strcmp(search->next[i]->name,string) == 0) && search->next[i]->type == 'd')
{
printf("\nDirectory name already used!");
```

```
goto skip4;
}
i++;
}
struct node *new = (struct node*) malloc(sizeof(struct node));
strcpy(new->name,string);
new->type = 'd';
new->prev = current;
for(i=0;i<10;i++)
{
new->next[i] = NULL;
}
search = current;
for(i=0;i<10;i++)
{
if(search->next[i] == NULL)
{
search->next[i] = new;
break;
}
}
skip4:
if(i == 10)
{
printf("\n%s directory is full!",current->name);
}
}
else
{
current = current->next[ch];
t_directory(current);
```

```
}
}
void t_file(struct node *current)
{
int i,ch;
char string[20];
printf("\nWhere do you want to create a file?");
if(current != head)
{
printf("\n-1.Here");
}
i = 0;
search = current;
while(i<10 && search->next[i] != NULL)
{
if(search->next[i]->type != 'f')
{
printf("\n%d.%s",i,search->next[i]->name);
}
i++;
}
printf("\n");
scanf("%d",&ch);
if(ch == -1)
{
printf("\nEnter name of file: ");
scanf("%s",string);
search = current;
i = 0;
while(i<10 && search->next[i] != NULL)
{
```

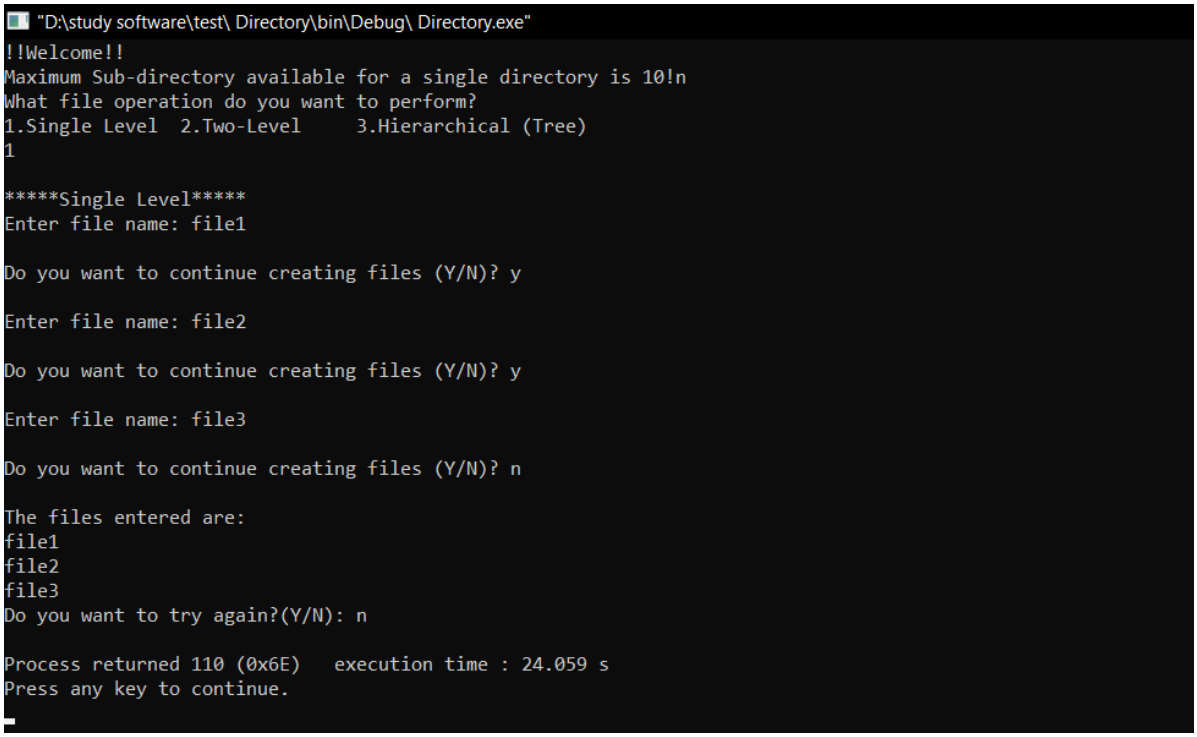
```
if((strcmp(search->next[i]->name,string) == 0) && search->next[i]->type == 'f')
{
printf("\nFile name already used!");
goto skip5;
}
i++;
}
struct node *new = (struct node*) malloc(sizeof(struct node));
strcpy(new->name,string);
new->type = 'f';
new->prev = current;
for(i=0;i<10;i++)
{
new->next[i] = NULL;
}
search = current;
for(i=0;i<10;i++)
{
if(search->next[i] == NULL)
{
search->next[i] = new;
break;
}
}
skip5:
if(i == 10)
{
printf("\n%s directory is full!",current->name);
}
}
else
```

```
{
current = current->next[ch];
t_file(current);
}
}
void tree()
{
int i,n;
char c;
printf("\n*****Hierarchichal*****");
struct node *new = (struct node*) malloc(sizeof(struct node));
strcpy(new->name,"root");
new->type = 'd';
new->prev = NULL;
for(i=0;i<10;i++)
{
new->next[i] = NULL;
}
head = new;
current = head;
do
{
printf("\n\nEnter your chioce: ");
printf("\n1.Create User\n2.Create Directory\n3.Create File\n");
scanf("%d",&n);
if(n == 1)
{
t_user(head);
}
else if(n == 2)
{
```

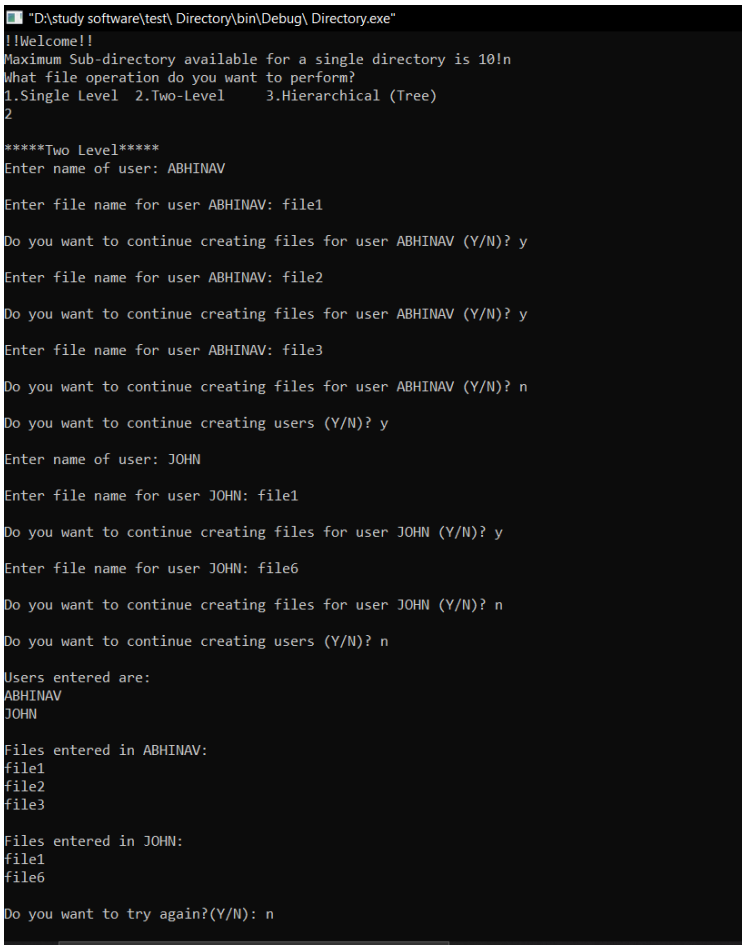
```
if(head->next[0] == NULL)
{
printf("\nEnter atleast one user!");
}
else
{
t_directory(head);
}
}
else if(n == 3)
{
if(head->next[0] == NULL)
{
printf("\nEnter atleast one user!");
}
else
{
t_file(head);
}
}
else
{
printf("\nOption not valid! Please choose again!");
}
printf("\nDo you want to continue operation on hierarchical file structure (Y/N)? ");
scanf(" %c",&c);
}
while(c == 'Y' || c =='y');
}
void main()
{
```

```
int ch;
char c;
printf("!!Welcome!!");
printf("\nMaximum Sub-directory available for a single directory is 10!\
n");
do
{
printf("\nWhat file operation do you want to perform?");
printf("\n1.Single Level\t2.Two-Level\t3.Hierarchical (Tree)\n");
scanf("%d",&ch);
if(ch == 1)
{
single_level();
}
else if(ch == 2)
{
two_level();
}
else if(ch == 3)
{
tree();
}
else
{
printf("\nOption not valid! Please choose again");
}
printf("\nDo you want to try again?(Y/N): ");
scanf(" %c",&c);
}
while(c == 'Y' || c =='y');
}
```

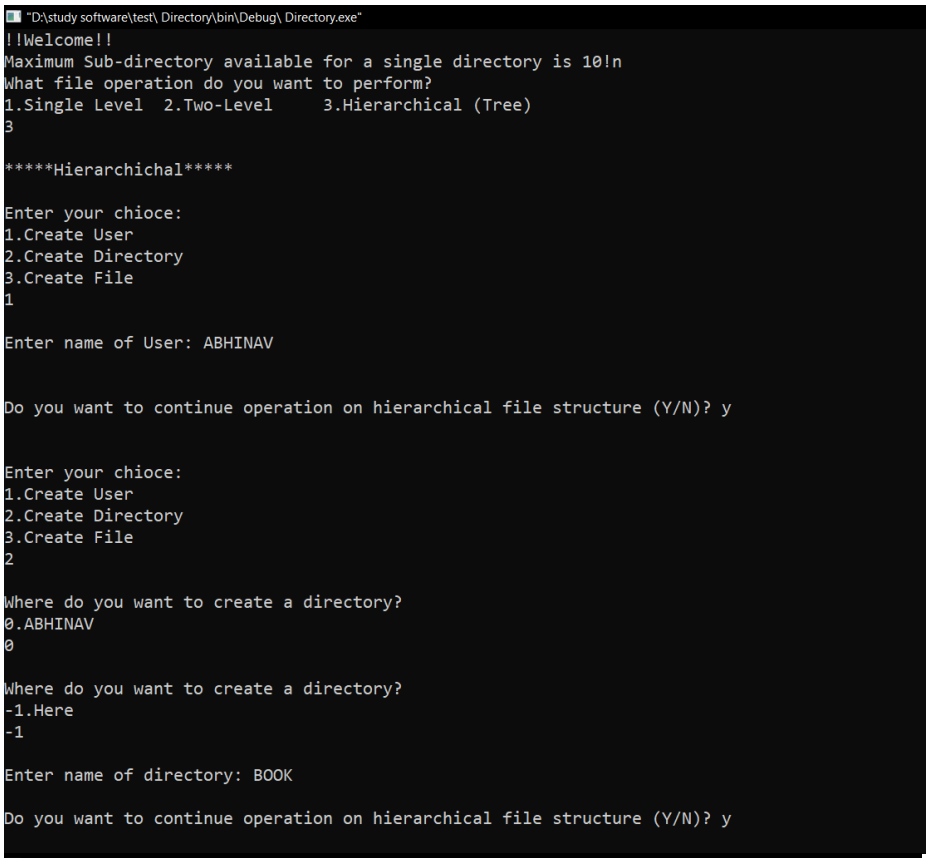
SINGLE LEVEL

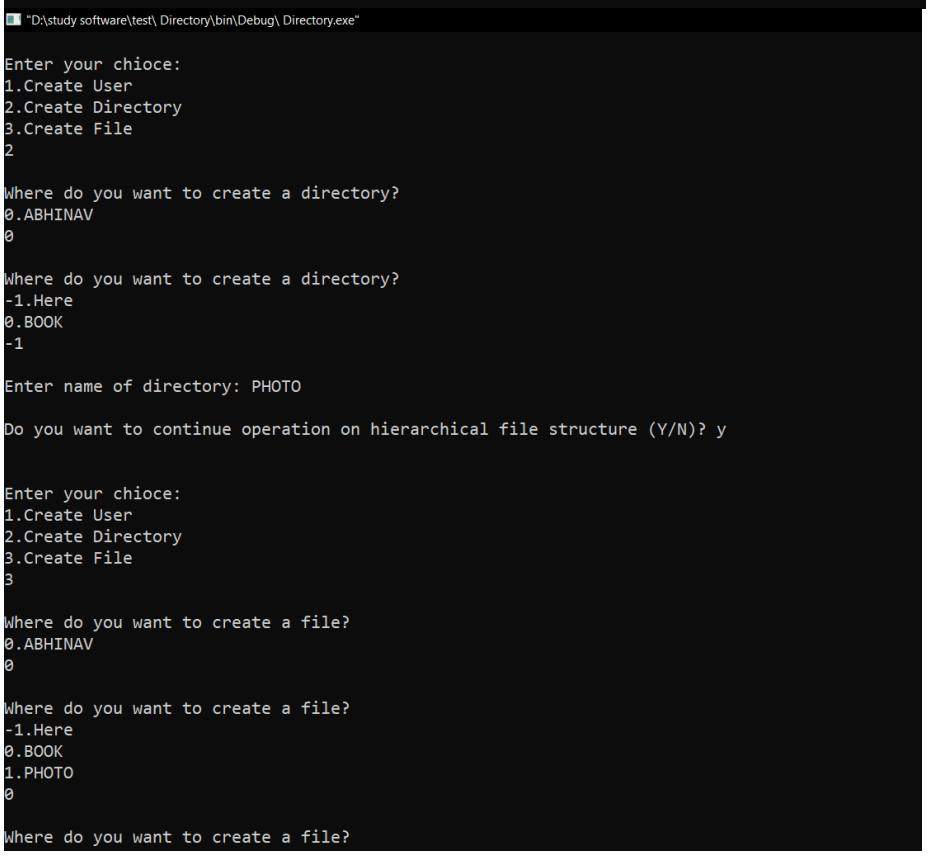


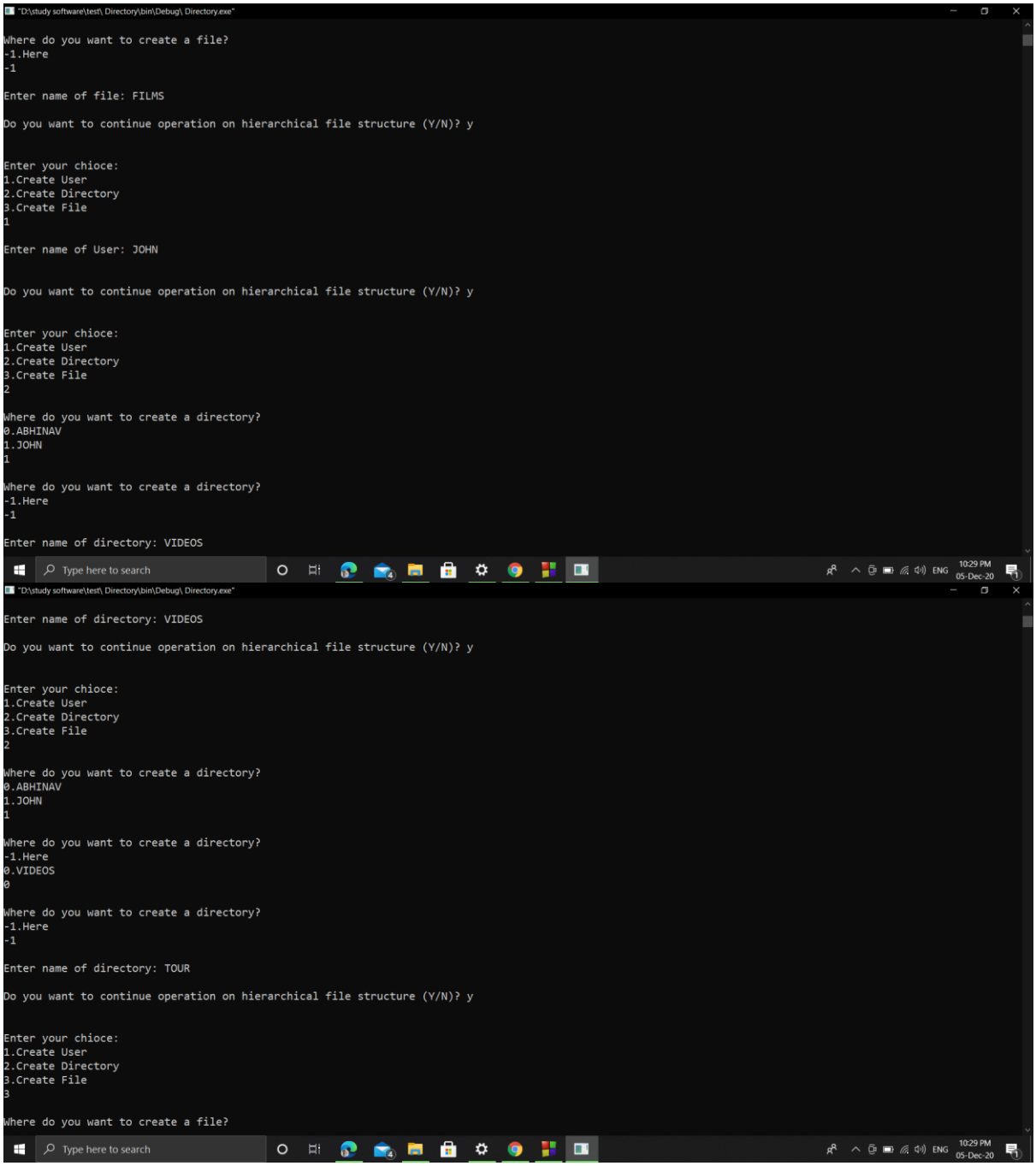
TWO-LEVEL

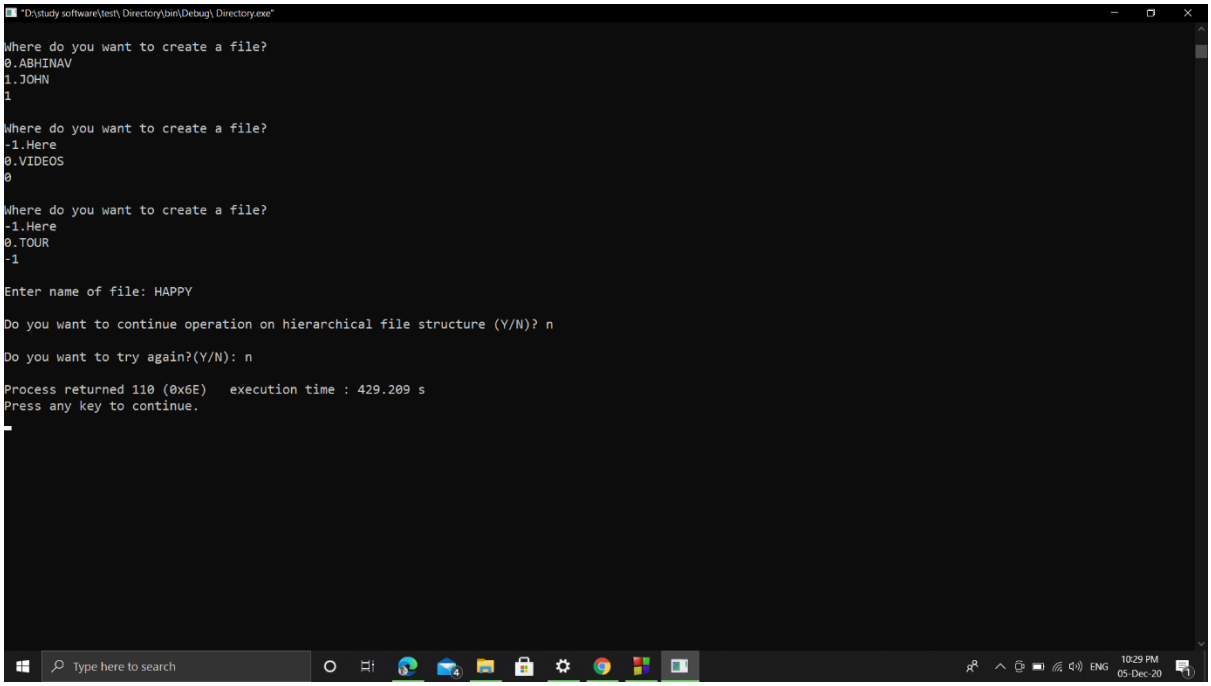


HIERARCHICAL






"D:\study software\test\ Directory\bin\Debug\ Directory.exe"
Where do you want to create a file?
-1.Here
-1
Enter name of file: FILMS
Do you want to continue operation on hierarchical file structure (Y/N)? y
Enter your chioce:
1.Create User
2.Create Directory
3.Create File
1
Enter name of User: JOHN
Do you want to continue operation on hierarchical file structure (Y/N)? y
Enter your chioce:
1.Create User
2.Create Directory
3.Create File
2
Where do you want to create a directory?
0.ABHINAV
1.JOHN
1
Where do you want to create a directory?
-1.Here
-1
Enter name of directory: VIDEOS
Type here to search
ENG
10:29 PM
05-Dec-20
"D:\study software\test\ Directory\bin\Debug\ Directory.exe"
Enter name of directory: VIDEOS
Do you want to continue operation on hierarchical file structure (Y/N)? y
Enter your chioce:
1.Create User
2.Create Directory
3.Create File
2
Where do you want to create a directory?
0.ABHINAV
1.JOHN
1
Where do you want to create a directory?
-1.Here
0.VIDEOS
0
Where do you want to create a directory?
-1.Here
-1
Enter name of directory: TOUR
Do you want to continue operation on hierarchical file structure (Y/N)? y
Enter your chioce:
1.Create User
2.Create Directory
3.Create File
3
Where do you want to create a file?
Type here to search
ENG
10:29 PM
05-Dec-20


"D:\study software\test\ Directory\bin\Debug\ Directory.exe"
Where do you want to create a file?
0.ABHINAV
1.JOHN
1
Where do you want to create a file?
-1.Here
0.VIDEOS
0
Where do you want to create a file?
-1.Here
0.TOUR
-1
Enter name of file: HAPPY
Do you want to continue operation on hierarchical file structure (Y/N)? n
Do you want to try again?(Y/N): n
Process returned 110 (0x6E)   execution time : 429.209 s
Press any key to continue.
Type here to search
ENG
10:29 PM
05-Dec-20