Dr. D.Y.Patil Arts, Commerce & Science College,
Pimpri, Pune-18
S.Y.B.Sc(Comp. Sci) 2024-25
Data Structures and Algorithms – I
Practical Assignment 2: Sorting Algorithms – Bubble Sort, Insertion Sort, Selection Sort
Set A

a) Sort a random array of n integers (accept the value of n from user) in ascending order by using bubble sort algorithm.

```
// Bubble sort on integer

#include<stdio.h>
void bubblesort(int [],int);
void display(int [],int);
main()
{
      int a[20],i,n;
      printf("\nHow many numbers:");
      scanf("%d",&n);
      printf("\nEnter the unsorted element:  ");
      for(i=0;i<n;i++)
            scanf("%d",&a[i]);
      bubblesort(a,n);
      printf("\nElements after sorting: ");
      display(a,n);
}
void bubblesort(int a[20],int n)
{
      int i,j,temp,pass;
      for(pass=1;pass<n;pass++)
      {
            for(i=0;i<n-pass;i++)
            {
                  if(a[i]>a[i+1])
                  {
                        temp=a[i];
                        a[i]=a[i+1];
                        a[i+1]=temp;

                  }
            }
      }
}

void display(int a[20],int n)
{
      int i;
      printf("\n");
      for(i=0;i<n;i++)
            printf("\t%d",a[i]);
}

/*

[root@localhost setA]# cc bubbleint.c
```

```
[root@localhost setA]# ./a.out

How many numbers:5

Enter the unsorted element:  4 7 9 2 6

Elements after sorting:
     2    4    6    7    9

*/
```

b) Sort a random array of n integers (create a random array of n integers) in ascending order by using insertion sort algorithm.

```
//insertion sort o integer
#include<stdio.h>
void insertsort(int [],int);
void display(int [],int);
main()
{
     int a[20],i,n;
     printf("\nHow many numbers:");
     scanf("%d",&n);
     printf("\nEnter the unsorted element:  ");
     for(i=0;i<n;i++)
          scanf("%d",&a[i]);
     insertsort(a,n);
     printf("\n\nThe Sorted list is: ");
     display(a,n);
}
void insertsort(int a[],int n)
{
     int i,j,key;
     for(i=1;i<n;i++)
     {
          key=a[i];

          for(j=i-1;j>=0 && a[j]>key;j--)
          {
               a[j+1]=a[j];
          }
          a[j+1]=key;
     }
}

void display(int a[20],int n)
{
     int i;
     for(i=0;i<n;i++)
          printf("\t%d",a[i]);
}
/*
   [root@localhost setA]# cc insertionint.c
   [root@localhost setA]# ./a.out

   How many numbers:5
```

```
   Enter the unsorted element:  7 2 1 5 3


   The Sorted list is:     1     2     3     5     7
  */
```

---

c) Sort a random array of n integers (accept the value of n from user) in ascending order by using selection sort algorithm.

```
//sekection sort on integer
#include<stdio.h>
void selectionsort(int a[20],int n);
main()
{
  int a[20],n,i;
  printf("\nenter total elements to store in array: ");
  scanf("%d",&n);
  printf("\nenter Unsorted data: ");
  for(i=0;i<n;i++)
  {
     printf("\nEnter %d num: ",i);
     scanf("%d",&a[i]);
  }
  selectionsort(a,n);
  printf("\nData After sorting: ");
  for(i=0;i<n;i++)
  {
      printf("   %d",a[i]);
  }

}
void selectionsort(int a[20],int n)
{
    int pass,key,i,temp;
    for(pass=0;pass<n-1;pass++)
    {
      key=pass;
      for(i=pass+1;i<n;i++)
      {
      if(a[key]>a[i])
      {
          temp=a[key];
          a[key]=a[i];
          a[i]=temp;
      }
      }
    }
}
/*
[root@localhost ass2]# cc selectionint.c
[root@localhost ass2]# ./a.out

enter total elements to store in array: 5
```

```
enter Unsorted data:
Enter 0 num: 45

Enter 1 num: 22

Enter 2 num: 78

Enter 3 num: 11

Enter 4 num: 36

Data After sorting:    11   22   36   45   78
*/
```

Set B

a) Read the data from the file "employee.txt" and sort on age using bubble sort, insertion sort and selection sort.

```
//bubble sort on age
#include<stdio.h>
#include<stdlib.h>
struct employee
{
      char ename[20];
      int age;
};
typedef struct employee emp;
int readfile(emp *);
void bubblesort(emp *,int);
void display(emp *,int);
main()
{
      int n;
      emp a[100];
      n=readfile(a);
         display(a,n);
      bubblesort(a,n);
         printf("\nData After Sorting: ");
      display(a,n);
}
int readfile(emp *a)
{
      char fname[20];
      int i=0;
      FILE *fp;
      printf("Enter file name: ");
      scanf("%s",fname);
      fp=fopen(fname,"r");
      if(fp==NULL)
      {
            printf("Error in opening File");
            exit(0);
      }
      while(!feof(fp))
```

```
      {
            fscanf(fp," %s %d",a[i].ename,&a[i].age);
            i++;
      }
      return i-1;
}
void bubblesort(emp *a,int n)
{
      int pass,i;
      emp temp;
      for(pass=1;pass<n;pass++)
      {
            for(i=0;i<n-pass;i++)
            {
                  if(a[i].age>a[i+1].age)
                  {
                        temp=a[i];
                        a[i]=a[i+1];
                        a[i+1]=temp;
                  }
            }
      }
}

void display(emp *a,int n)
{
      int i;
      for(i=0;i<n;i++)
            printf("\n%s %d",a[i].ename,a[i].age);
}
/*
employee.txt
fff 34
bbb 29
aaa 22
ccc 58
eee 37
ddd 41

[root@localhost ass2]# cc bubbleage.c
[root@localhost ass2]# ./a.out
Enter file name: employee.txt

fff 34
bbb 29
aaa 22
ccc 58
eee 37
ddd 41
Data After Sorting:
aaa 22
bbb 29
fff 34
eee 37
ddd 41
ccc 58
```

```
*/
```

```
//bubble sort on age with write file function
#include<stdio.h>
#include<stdlib.h>
struct employee
{
      char ename[20];
      int age;
};
typedef struct employee emp;
emp e[100];
int readfile();
void bubblesort(int);
void display(int);
void writefile(int);
void main()
{
      int n;
      emp e[100];
      n=readfile();
      bubblesort(n);
      display(n);
      writefile(n);
}
int readfile()
{
      char fname[20];
      int a,i=0;
      FILE *fp;
      printf("enter file name");
      scanf("%s",fname);
      fp=fopen(fname,"r");
      if(fp==NULL)
      {
            printf("Error in opening File");
            exit(0);
      }
      while(!feof(fp))
      {
            fscanf(fp,"%s %d",e[i].ename,&e[i].age);
            printf("\n%s %d",e[i].ename,e[i].age);
            i++;
      }
      return i-1;
}
void bubblesort(int n)
{
      int pass,i,j;
      emp temp;
      for(pass=1;pass<n;pass++)
      {
            for(j=0;j<n-pass;j++)
            {
                  if(e[j].sal>e[j+1].sal)
                  {
```

```
                        temp=e[j];
                        e[j]=e[j+1];
                        e[j+1]=temp;
                    }
                }
        }
}

void display(int n)
{
        int i;
        for(i=0;i<n;i++)
                printf("\n%s %d",e[i].ename,e[i].age);
}
void writefile(int n)
{
        char fname[20];
        int a,i=0;
        FILE *fp;
        printf("\nEnter file name");
        scanf("%s",fname);
        fp=fopen(fname,"w");
        if(fp==NULL)
        {
                printf("error");
                exit(0);
        }
        for(i=0;i<n;i++)
        {
                fprintf(fp,"\n%s %d %d",e[i].ename,e[i].age);
                //printf("\n%s%d%d",e[i].ename,e[i].age,e[i].sal);

        }

}
```

```
//insertion sort on age
#include<stdio.h>
#include<stdlib.h>
struct employee
{
        char ename[20];
        int age;
        int sal;
};
typedef struct employee emp;
int readfile(emp *);
void insertionsort(emp *,int);
void display(emp *,int);
main()
{
        int n;
        emp a[100];
        n=readfile(a);
        printf("\nStructure data after reading the file:");
        display(a,n);
```

```
      insertionsort(a,n);
      printf("\nStructure data after sorting");
      display(a,n);
}
int readfile(emp *a)
{
      char fname[20];
      int i=0;
      FILE *fp;
      printf("\nEnter file name to read:  ");
      scanf("%s",fname);
      fp=fopen(fname,"r");
      if(fp==NULL)
      {
            printf("\nError in opening file!");
            exit(0);
      }
      while(!feof(fp))
      {
            fscanf(fp,"%s %d",a[i].ename,&a[i].age);
            i++;
      }
      return i-1;
}
void insertionsort(emp *a,int n)
{
      int i,j;
      emp key;
      for(i=1;i<n;i++)
      {
            key=a[i];

            for(j=i-1;j>=0 && a[j].age>key.age;j--)
            {
                  a[j+1]=a[j];
            }
            a[j+1]=key;
      }
}

void display(emp *a,int n)
{
      int i;
      for(i=0;i<n;i++)
            printf("\n%s %d",a[i].ename,a[i].age);
}
/*
[root@localhost ass2]# cc insertionage.c
[root@localhost ass2]# ./a.out

Enter file name to read:  employee.txt

Structure data after reading the file:
fff 34
bbb 29
aaa 22
```

```
ccc 58
eee 37
ddd 41
Structure data after sorting
aaa 22
bbb 29
fff 34
eee 37
ddd 41
ccc 58
*/
```

```
//selection sort on age
#include<stdio.h>
#include<stdlib.h>
struct employee
{
      char ename[20];
      int age;
};
typedef struct employee emp;
int readfile(emp *);
void selectionsort(emp *,int);
void display(emp *,int);
main()
{
      int n;
      emp a[100];
      n=readfile(a);
         display(a,n);
      selectionsort(a,n);
         printf("\nData After Sorting: ");
      display(a,n);
}
int readfile(emp *a)
{
      char fname[20];
      int i=0;
      FILE *fp;
      printf("Enter file name: ");
      scanf("%s",fname);
      fp=fopen(fname,"r");
      if(fp==NULL)
      {
            printf("Error in opening File");
            exit(0);
      }
      while(!feof(fp))
      {
            fscanf(fp," %s %d",a[i].ename,&a[i].age);
            i++;
      }
      return i-1;
}
void selectionsort(emp *a,int n)
{
```

```
    int pass,key,i;
    emp temp;
    for(pass=0;pass<n-1;pass++)
    {
      key=pass;
      for(i=pass+1;i<n;i++)
      {
         if(a[key].age>a[i].age)
         {
             temp=a[key];
             a[key]=a[i];
             a[i]=temp;
         }
      }
    }
}


void display(emp *a,int n)
{
     int i;
     for(i=0;i<n;i++)
          printf("\n%s %d",a[i].ename,a[i].age);
}
/*
[root@localhost ass2]# cc selectionage.c
[root@localhost ass2]# ./a.out
Enter file name: employee.txt

fff 34
bbb 29
aaa 22
ccc 58
eee 37
ddd 41
Data After Sorting:
aaa 22
bbb 29
fff 34
eee 37
ddd 41
ccc 58
*/
```

---

b) Read the data from the file “employee.txt” and sort on names in alphabetical order (use strcmp) using bubble sort, insertion sort and selection sort.

```
// bubble sort on names
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
struct employee
{
     char ename[20];
     int age;
```

```
};
typedef struct employee emp;
int readfile(emp *);
void bubblesort(emp *,int);
void display(emp *,int);
main()
{
      int n;
      emp a[100];
      n=readfile(a);
      printf("\nStructure data after reading from file:");
      display(a,n);
      bubblesort(a,n);
      printf("\nStructure data after sorting: ");
      display(a,n);
}
int readfile(emp *a)
{
      char fname[20];
      int i=0;
      FILE *fp;
      printf("\nEnter file name to read: ");
      scanf("%s",fname);
      fp=fopen(fname,"r");
      if(fp==NULL)
      {
            printf("\nError in opening file!");
            exit(0);
      }
      while(!feof(fp))
      {
            fscanf(fp,"%s %d",a[i].ename,&a[i].age);
            i++;
      }
      return i-1;
}
void bubblesort(emp *a,int n)
{
      int pass,i,j;
      emp temp;
      for(pass=1;pass<n;pass++)
      {
            for(j=0;j<n-pass;j++)
            {
                  if((strcmp(a[j].ename,a[j+1].ename)>0))
                  {
                        temp=a[j];
                        a[j]=a[j+1];
                        a[j+1]=temp;
                  }
            }
      }
}

void display(emp *a,int n)
{
```

```
      int i;
      for(i=0;i<n;i++)
            printf("\n%s %d",a[i].ename,a[i].age);
}
/*
[root@localhost ass2]# cc bubblename.c
[root@localhost ass2]# ./a.out

Enter file name to read: employee.txt

Structure data after reading from file:
fff 34
bbb 29
aaa 22
ccc 58
eee 37
ddd 41
Structure data after sorting:
aaa 22
bbb 29
ccc 58
ddd 41
eee 37
fff 34
*/
```

```
//insertion sort on names
#include<stdio.h>
#include<stdlib.h>
struct employee
{
      char ename[20];
      int age;
};
typedef struct employee emp;
int readfile(emp *);
void insertionsort(emp *,int);
void display(emp *,int);
main()
{
      int n;
      emp a[100];
      n=readfile(a);
      printf("\nStructure data after reading from file: ");
      display(a,n);
      insertionsort(a,n);
      printf("\nStructure data after sorting: ");
      display(a,n);
}
int readfile(emp *a)
{
      char fname[20];
      int i=0;
      FILE *fp;
      printf("\nEnter file name to read:  ");
      scanf("%s",fname);
```

```
      fp=fopen(fname,"r");
      if(fp==NULL)
      {
            printf("\nError in opening file!");
            exit(0);
      }
      while(!feof(fp))
      {
            fscanf(fp,"%s %d",a[i].ename,&a[i].age);
            i++;
      }
      return i-1;
}
void insertionsort(emp *a,int n)
{
      int i,j;
      emp key;
      for(i=1;i<n;i++)
      {
            key=a[i];

            for(j=i-1;j>=0 && (strcmp(a[j].ename,key.ename)>0);j--)
            {

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

void display(emp *a,int n)
{
      int i;
      for(i=0;i<n;i++)
            printf("\n%s %d",a[i].ename,a[i].age);
}
/*
[root@localhost ass2]# cc insertionname.c
[root@localhost ass2]# ./a.out

Enter file name to read:  employee.txt

Structure data after reading from file:
fff 34
bbb 29
aaa 22
ccc 58
eee 37
ddd 41
Structure data after sorting:
aaa 22
bbb 29
ccc 58
ddd 41
eee 37
fff 34
```

*/

```
//selection sort on name
#include<stdio.h>
#include<stdlib.h>
struct employee
{
      char ename[20];
      int age;
};
typedef struct employee emp;
int readfile(emp *);
void selectionsort(emp *,int);
void display(emp *,int);
main()
{
      int n;
      emp a[100];
      n=readfile(a);
         display(a,n);
      selectionsort(a,n);
         printf("\nData After Sorting: ");
      display(a,n);
}
int readfile(emp *a)
{
      char fname[20];
      int i=0;
      FILE *fp;
      printf("Enter file name: ");
      scanf("%s",fname);
      fp=fopen(fname,"r");
      if(fp==NULL)
      {
            printf("Error in opening File");
            exit(0);
      }
      while(!feof(fp))
      {
            fscanf(fp," %s %d",a[i].ename,&a[i].age);
            i++;
      }
      return i-1;
}
void selectionsort(emp *a,int n)
{
    int pass,key,i;
    emp temp;
    for(pass=0;pass<n-1;pass++)
    {
      key=pass;
      for(i=pass+1;i<n;i++)
      {
         if(strcmp(a[key].ename,a[i].ename)>0)
         {
```

```
            temp=a[key];
            a[key]=a[i];
            a[i]=temp;
         }
      }
    }
}


void display(emp *a,int n)
{
     int i;
     for(i=0;i<n;i++)
          printf("\n%s %d",a[i].ename,a[i].age);
}
/*
[root@localhost ass2]# cc selectionname.c
[root@localhost ass2]# ./a.out
Enter file name: employee.txt

fff 34
bbb 29
aaa 22
ccc 58
eee 37
ddd 41
Data After Sorting:
aaa 22
bbb 29
ccc 58
ddd 41
eee 37
fff 34
*/
```