

# Sorting (Bubble)

In this section of notes you will learn one technique for ordering a list

James Tam

Algorithm: Original List

E	A	D	B	A
---	---	---	---	---

James Tam

### Algorithm: First Pass Through The List

Start 

E	A	D	B	A
---	---	---	---	---

1<sup>st</sup> swap 

A	E	D	B	A
---	---	---	---	---

2<sup>nd</sup> swap 

A	D	E	B	A
---	---	---	---	---

3<sup>rd</sup> swap 

A	D	B	E	A
---	---	---	---	---

4<sup>th</sup> swap 

A	D	B	A	E
---	---	---	---	---

James Tam

### Algorithm: Second Pass Through The List

Start 

A	D	B	A	E
---	---	---	---	---

1<sup>st</sup> swap 

A	B	D	A	E
---	---	---	---	---

2<sup>nd</sup> swap 

A	B	A	D	E
---	---	---	---	---

James Tam

### Algorithm: Third Pass Through The List

Start 

A	B	A	D	E
---	---	---	---	---

1<sup>st</sup> swap 

A	A	B	D	E
---	---	---	---	---

James Tam

### Algorithm: Fourth Pass Through The List

Start 

A	A	B	D	E
---	---	---	---	---

James Tam

## A Sorting Example

A complete version of this example program can be found under  
/home/231/examples/sorting/investors.p

```
program investors (input, output);

const
  MAX_CLIENTS      = 10;
  NAME_LENGTH      = 30;
  EMAIL_LENGTH     = 30;
  FILE_NAME_LENGTH = 256;

type
  Client = record
    firstName : string [NAME_LENGTH];
    lastName  : string [NAME_LENGTH];
    income    : real;
    email     : string [EMAIL_LENGTH];
  end; (* Declaration of record client *)

  ClientList = array [1..MAX_CLIENTS] of Client;
```

James Tam

## A Sorting Example (2)

```
procedure readClientInformation (var aClientList : ClientList ;
                                var clientCount : integer);

var
  investorData : text;
  inputFileName : string[FILE_NAME_LENGTH];
begin;
  writeln;
  write('Enter the name of the input file: ');
  readln(inputFileName);
  reset(investorData, inputFileName);
  writeln('Opening file ', inputFileName, ' for reading');
  clientCount := 0;
  if EOF (investorData) then
  begin
    writeln('File ', inputFileName, ' is empty: nothing to read.');  end
```

James Tam

## A Sorting Example (3)

```
else
begin
  while NOT EOF (investorData) AND (clientCount < MAX_CLIENTS) do
  begin
    clientCount := clientCount + 1;
    with aClientList[clientCount] do
    begin
      readln(investorData, firstName);
      readln(investorData, lastName);
      readln(investorData, income);
      readln(investorData, email);
      readln(investorData);  (* Eat the newline between records *)
    end;
  end; (* while *)
end; (* else *)
close(investorData);
end; (* End of procedure readClientInformation *)
```

James Tam

## A Sorting Example (4)

```
procedure displayInstructions;
begin
  writeln;
  writeln('This program allows you track a list of clients each of which is');
  writeln('is an investor. The initial investor information will be read');
  writeln('from the file called "investorList". From there you can display');
  writeln('the list of clients onscreen, add a client, modify the');
  writeln('information for a client, erase clients from the list, recorder');
  writeln('the list of clients or search for a client. When you are done');
  writeln('quit the program and all of your changes will be written out to');
  writeln('an output file.');
```

```
end;
```

James Tam

## A Sorting Example (5)

```
procedure displayMenu;
begin
  writeln;
  writeln('Options');
  writeln('(d)isplay list of clients');
  writeln('(a)dd a new client');
  writeln('(e)rase a client"s record from the list');
  writeln('(m)odify an existing client');
  writeln('(r)eorder the client list alphabetically by last name');
  writeln('(s)earch for a client');
  writeln('(q)uit the program. ');
  write('Enter your selection: ');
end; (* End of procedure displayMenu *)
```

James Tam

## A Sorting Example (6)

```
procedure displayClientList (aClientList : ClientList;
                             clientCount : integer );
var
  i : integer;
begin
  writeln;
  writeln('DISPLAYING CLIENT LIST');
  writeln('-----');
  if (clientCount = 0) then
    writeln('No clients in the list: nothing to display');
```

James Tam

## A Sorting Example (7)

```
for i := 1 to clientCount do
begin
  with aClientList[i] do
  begin
    writeln('First name: ', firstName);
    writeln('Last name: ', lastName);
    writeln('Income :$', income:0:2);
    writeln('Email: ', email);
    end; (*with-do *)
    writeln;
  end; (* for *)
end; (* End of procedure displayClientList *)
```

James Tam

## A Sorting Example (8)

```
procedure addClient (var aClientList : ClientList;
                    var clientCount : integer);

var
  newClient : Client;
begin
  writeln;
  writeln('Adding new client to list of clients');
```

James Tam

## A Sorting Example (9)

```
with newClient do
begin
  write('Enter first name of client (max 30 characters): ');
  readln(firstName);
  write('Enter last name of client (max 30 characters): ');
  readln(lastName);
  write('Enter annual gross income of client (max 8 digits with no commas)
      $');
  readln(income);
  write('Enter email of client (max 30 characters):');
  readln(email);
  writeln;
end; (* with-do *)
clientCount := clientCount + 1;
aClientList[clientCount] := newClient;
writeln('Added new client ', newClient.lastName);
end; (* End of procedure addClient *)
```

James Tam

## A Sorting Example (10)

```
procedure swap (var first : Client;
               var second : Client);
var
  temp : Client;
begin
  temp := first;
  first := second;
  second := temp;
end; (* End of procedure swap *)
```

James Tam



## A Sorting Example (11)

```
procedure reorder (var aClientList : ClientList;
                  clientCount : integer);
var
  i      : integer;
  isSorted : boolean;
begin
```

James Tam

## A Sorting Example (12)

```
repeat
begin
  (* Assume list is sorted until proven otherwise. *)
  isSorted := True;
  for i := 1 to (clientCount-1) do
  begin
    if (aClientList[i].lastName > aClientList[i+1].lastName) then
    begin
      swap(aClientList[i], aClientList[i+1]);
      isSorted := False;
    end; (* if-then *)
  end; (* for *)
end; (* repeat-until *)
until (isSorted = True);
end; (* End of procedure reorder *)
```

James Tam

## A Sorting Example (13)

```
procedure saveClientInformation (aClientList : ClientList;
                                clientCount : integer);
var
  i                : integer;
  upDatedInvestorData : text;
  outputFileName   : string[FILE_NAME_LENGTH];
begin
  writeln;
  write('Enter the name of the file to save client information to: ');
  readln(outputFileName);
  rewrite(updatedInvestorData, outputFileName);
  writeln('Saving updated investor information to file ', outputFileName);
```

James Tam

## A Sorting Example (14)

```
for i := 1 to clientCount do
begin
  with aClientList[i] do
  begin
    writeln(updatedInvestorData, firstName);
    writeln(updatedInvestorData, lastName);
    writeln(updatedInvestorData, income:0:2);
    writeln(updatedInvestorData, email);
    writeln(updatedInvestorData);
  end;
end;
writeln(updatedInvestorData);
close(updatedInvestorData);
end; (* End of procedure saveClientInformation *)
```

James Tam

## A Sorting Example (15)

```
procedure processMenuSelection (var aClientList : ClientList;
                               var clientCount : integer;
                               menuSelection : char);
begin
  case (menuSelection) of
    'D', 'd' :
      begin
        displayClientList(aClientList, clientCount);
      end;

    'A', 'a' :
      begin
        addClient(aClientList, clientCount);
      end;
  end;
```

James Tam

## A Sorting Example (16)

```
'E', 'e' :
begin
  writeln;
  writeln('Erase client record: You need to write the code to do');
  writeln('this in your own program.');
```

```
'M', 'm' :
begin
  writeln;
  writeln('Modify client: You need to write the code to do this');
  writeln('in your own program.');
```

```
'R', 'r' :
begin
  writeln;
  reorder (aClientList, clientCount);
end;
```

James Tam

## A Sorting Example (17)

```
'S', 's' :  
begin  
  writeln;  
  writeln('Search for a client: You need to write the code to do');  
  writeln('this in your own program.');
```

  

```
'Q', 'q' :  
begin  
  writeln;  
  writeln('Thank you for using the investor 2000 (TM) program.');
```

  

```
  writeln('Come again!');  
  writeln;  
end;
```

James Tam

## A Sorting Example (18)

```
else  
begin  
  writeln;  
  writeln('Please enter one of the following options: "d","a",  
    "e","m","r","s" or "q"');
```

  

```
  writeln;  
end;  
end; (* Case with menu options*)  
end; (* End of processMenuSelection *)
```

James Tam

## A Sorting Example (19)

```
(*-----*)
                          Start of main program
(*-----*)
begin
  var tamjClientList : ClientList;
  var clientCount   : integer;
  var menuSelection : char;

  readClientInformation(tamjClientList, clientCount);
  displayInstructions;
  repeat
  begin
    displayMenu;
    readln(menuSelection);
    processMenuSelection(tamjClientList,clientCount,menuSelection);
  end; (* repeat-until loop *)
  until (menuSelection = 'Q') OR (menuSelection = 'q');

  saveClientInformation(tamjClientList, clientCount);
end. (*           End of main program           *)
```

James Tam

## You Should Now Know

- How to trace through a program that employs the Bubble Sort
- How to write the code for a bubble sort to reorder an array

James Tam