Source 🡪 object 🡪 executable

Source files

**Preamble**

-#include

-Function prototype

-global variable

-new structures

**Body**

-function definition

Code block

Make a new console project

Output is with stream

-console use cout

-files use ofstream

//basic file

#include <iostream>  
#include <fstream>

Using namespace std;

Int main()

{

Ofstream outfile;//open file

Outfile.open(“test txt”, ofstream :: out);

Outfile <<”send”<<endl;

Outfile<<”more”<<endl;

//close file

Outfiles .close();

Return 0;

}

Variable

Information that is tracked in a program must be through variables. Variable are created at the lines in the program and destroyed at the end of their braces.

{

Int a;

Cout<<””;

{

Int b;

Cout<<””;

}//b destroyed

Return 0;

}//a destroyed

Built in type

Bool, charm short, int, long, long long, float, double.

String is a compound type.

Feb 6 2013

Conditional

* To see if a condition is true
* Two ways to get conditional:
  + If statement OR Switch

Switch(\*variable)  
{

Case 3:

Break;

Case 5:

Break;

Default

}

If(true) if(!false if(1) //always true

If(false) if(!true) if(0) //always false

|| = or

&& = and

== is comparing

!= not equal to

Feb 7

Two numbers (a,b)

G = gcd(a,b) 24/18 = 1 24%18= 6 🡨 gcd

L = lcm(a,b) 18/6 = 3 18%6 = 0

Gl = ab

Looping

For(initial, condition, finial)//set amount of loops

While(condition)//un-set amount of loops

Do

{}

While(condition) //like while, but loop at least once

GCD – greatest common divisor

Euclidian algorithm

210/45 = 4 210%45 = 30

45/30 = 1 45%30 = 15 GCD (210,45) = 15

30/15 30%15 = 0

Feb 12

Alternating harmonic series

H = ∞∑k=1(-1)k+1/k =1/1 - ½ + 1/3 – ¼ =ln2

Write a program that sums the first n terms of the AHS

The naïve approach has a loop that starts at k=1, goes to n and adds terms each time

To accurately sum the series, we loop needs to start at k=n, and going to k=1

Feb13

String library

#include<string>

Strings are not built in type for c++. They are arrays of characters, calculated in double quotes;

“this is a string”//string

“”//empty string

‘n’//char, not string

Input with string

* Cin, one word at a time
* Getline, a line at a time

#include<cctype>//for character type

isUpper toUpper

isLower toLower

isDigit

To convert int to string or string to int

We use stringstream.

//Int to string

#include<sstream>

{

Stringstream ss;

Int num = 1000;

String str;

Ss << nums;

Str = ss.str();

}

//String to int

{String numString = “1000”;

Int num;

Stringstream ss;

Ss << numString;

Ss >> nums;}

Write a program that list all the word that start or end with the letter a.

Feb 14

Anagrams

-a rearrangement of a word or phrase that produces another word or phrase

Write a program that determines if two phrases are anagram.

-input strings

-count up all characters, ‘a’= ‘b’= ‘c’=

-if other string is the same, exist anagrams

Feb 15

Permutation

-is a rearrangement of a sequence

-the simplest way to list permutations is to use recursion(unit 2)

Write a program that lists all permutations for the numbers from 1 to n.

-this solution is iterative instead (uses loops)

Consider all permutations of the number 1, 2, 3

3 \* 2 \* 1 = 6(the amount of number after elimination) permutation

123 132 213 231 312 321

Permutation of 1-4 = 4\*3\*2\*1 = 24

\*make a number spiral

Start at 1 and write a spiral to the input number.

\*need to be right-justified

Feb20

7 1 6

7+1+6 = 14 🡪 1+4 = 5

Digital additive root

Feb25

Review of data structures

A struct in C++ is an object that allows two or more variable to be considered as a single type.

Consider making a console-based graph.

Make a structure for points

Struct point//struct name

{

Double x, y;//member variables

};

Now a point is usable in the code like another type.

To access member variables, use the dot operator.

Point pt;

Pt.x =1.3;

Pt.y=-4.5//pt=(1.3, -4.5)

Point pt2;

Pt2 = pt;//pt2 = (1.3, -4.5)

Pointers

* points to a memory locations
* int i=5;//creates an integer and gives it the value 5
* int \*p;//point to a memory location that stores an int
* p=&I;//p points to the memory location of integer i//& 🡨 gets the address of i
* \*p=2;//the value of the memory location at p is now 2//\*p changes the value of p
* Cout<<i<<endl;//prints I, which is now 2
* Pointers are declared as type \*name;
  + A statement is written with &
  + A value of… \*

Int i=10;

Int\*p, \*q;

P=&I;//p==10

Q=p;//q==10

I=20;//\*p=\*q==20

Pointers an point to arrays

Int arr[5]={1,2,3,4,5};

Int \*p;

P=arr;//p=&arr[0]

int val=p[3];//val =4

pointers are used to pass data back from functions.

Linked list

A linked list is one of the 5 fundamental data structures in computer science. Escaps, linked list, trees, stacks, and queues.

Operation arrays linked lists

Acces O(1)\*order # of permutation O(n)

Insertion O(n) O(1)

Deletion O(n) O(1)

Append O(n) O(1)

Prepend O(n) O(1)

A linked list is a structure that contains a data packet and a pointer to the next node in the list.

//single linked list

Struct node

{ int data;

Node \*next; };

//double linked list

Struct node

{ int data;

Node \*previous;

Node \*next; };

//circular linked list

Struct node

{ int data;

Node \*next; };

Recursion

Recursion is dividing a large problem into smaller similar problems that are easier to solve each of these little problems is then solved trivially and the overall problem can be solved by resembling the pieces.

Fibonacci sequence

This is a sequence defined recursively

Fn= fn-1 + fn-2

To solve this, save all previously calculated values of each Fibonacci number.

O(2n) = O(n)

Doubly linked list(assignment #5)

Struct node

{ int data;

Node \*next;

Node \*previous;

};

Int main()

{ ……

Node \*current, \*begin;

Current = new nodel

Current -> data = 1;

Current ->next=NULL;

Current->previos=NULL;

Begin = current;

…..

}

Void printBackwards

{ while(node!=NULL)

{ …

Node = node-> prev;

}

….

}

Append (assign #5 cont)

Node -> next = newnode;

Newnode -> next = NULL;

Newnode ->prev = node;

Return newnode;

Insert (#5)

Newnode -> prev = node;

Newnode -> next=node->next;

Node->next -> prev = newnode;

Node -> next = newnode;

Return newnode;

If(n==1)

Return 1;

Else

Int Value=f(n-1)\*n

Cout<<value;

Return val;

March 4

Stacks

A stack in computer science is a collection of objects that are accessed through pushing to the top of the stack and popping off from the top of the stack. Stacks are used for managing variables and function when programming. ||||||||🡪popping

🡪Pushing

//dummy program

Void func()

{int I = 4;}

Int main()

{int i = 3;

{int i=2;

Func();}

Func();

Return0;

}

* Last in First out

Stacks are often implemented using linked lists.

QUEUES

-dequeue(pop)

-enqueue(add, from the back) enqueue🡪 |||||||🡪dequeue

Queues are used for scheduling. This is for messages, processing for multiprocessors, auctioning. Queues are often implemented using doubly linked lists.

Tower of Hanoi

Tower(source, destination, storage)

* Tower(source, storage, destination)
  + Tower(storage, destination, source)

Greatest Common divisors

For a,b (- Z+🡨 positive integer

Gcd (a,b) efficiently, use Euclidean algorithm

R-2 = max(a,b)

R-1= min(a,b)

R(n-2)=q(n)R(n-1)+R(n) 🡪 GCD (a,b) = (q(k),r(k))

Q(K) = |r(k-2)/r(k-2)|

R(K) = R(K-2)%R(K-1)

March 18

Intro to trees

A tree is a data structure that is a node that contains a collection of pointers to children nodes. Trees are used to solve problems involving hierarchies.

The simplest type of tree is a binary tree. Each nodes points to 0, 1 or 2 child nodes.

A binary tree uses data structures:

Struct node

{

Int data;

Node \*left;

Node \*right;

};

Often tree nodes will have a parent node pointer.

In order to manipulate trees, recursion is (almost) always required.

Fundamental operations on trees are:

Inserting elements, deleting elements, finding values, finding max/min elements, traversing the tree

Traversing a Tree

Most operations with trees involve traversing (or walking) the tree. This takes O(log n) operations typically. As an example, to insert data into a tree, takes O(logn), and for n items, this takes O(nlogn). Traversing the tree (to print it out) is O(n). This means that sorting using a binary tree is O(nlogn).

Another example of traversing a tree is search. To find a data element in a tree takes O(logn) operations.

Implement a binary search using trees.

Minesweeper

\*This is an example of finding all connected cells. This is a recursive algorithm.

Dynamic programing is a method where the program keeps track of existing calculations so those calculations are not repeated.

This is used most often with recursion. In class, this was done with recursive FIB calculations.

Graph

In computer Science, the word graph has multiple meanings. One of the meanings is:

A graph is a collection of nodes and edges

\*facebook

Every person is a node to the graph and friend links are align in the group

Common question that can be asked about graphs

-how many connections does this person have? 🡪Add direct links

-is there a connection between these two people 🡪Find a path

-I want to add/delete this person 🡪

-what is the shortest path between

Graphs are either sparse or dense

Dense graphs have n edges for almost all n nodes. (Therefore a n2 edges)

Sparse graphs have very few edges

Sparse graph are implemented using a node structure where each node contains a list of pointers to neighbouring nodes.

Struct node

{

Int data;

Vector <node \*> nodes;

};

To find path between nodes, start with the first node “A”. Make “A” the root of a tree. All connection between “A” and its neighbours become children of “A”. Also mark each of these nodes as having been visited once. Continue this process until the entire graph is exhausted or “B” is found.

Sorting

Naïve or basic sorting algorithm tend to be O(n2) where n is the number of elements, and efficient sorting algorithms tend to be O(nlogn). One of the simplest sorting method is called insertion sort and is O(n2).

Consider the data set:

12 4 3 1 6 8

12

4 12

3 4 12

1 3 4 12

1 3 4 6 12

1 3 4 6 8 12

Merge sort

The merge sort is a “divide-and-merge” algorithm. This means the original data set is divided recursively into a collection of data sets that are easier to sort. Then these sets are conquered by merging them together recursively in a sorted way.

L1:12 4 5 17 ||R1: 3 2 8 6

L2:12 4 ||R2: 5 17 ||L3: 3 2 ||R3: 8 6

L4:12 ||R4:4 ||L5:5 ||R5:17 ||L6:3 ||R6:2 ||L7:8 ||R7:6

L8:4 5 12 17 R8:2 3 6 8

2 3 4 5 6 8 12 17

Eight Queen Problem April 2, 2013

Place eight queens on a standard 8x8 chessboard so that no queens are attacking each other.

What are all the possible solutions of the eight queen problem?

There are many ways to solve the eight queen problem. We will use backtracking.

Fractals/L-system

A fractal is a self-similar mathematical object.

Self-similar means that the object looks the same/similar at near distances and fat distances.

Fractals are primarily developed with some initial configuration and a set of simple recursive rules.

L-systems are fractals that operate on strings and are used to model some biological system.

Consider the string with only 2 possible letters A and B.

An initial configuration A

Rules A🡪AB

B🡪A

N=0 A

N=1 A B

N=2 A B A

N=3 A B A A B

Shortest distance for graphs

Dijkstra’s algorithm is one of the simpler methods for finding the shortest distance between two vertices in a graph. Start at the first vertex, find all adjacent vertices. Then consider each vertex and find all of their adjacent vertices, and so on. This continues until all vertices had been reached. The final algorithm then has a list of the shortest distance from any vertex from any other vertex. A list must be maintained so vertices are only visited once. Anytime a vertex is encountered with a smaller distance to the start node, use the smaller distance instead.

1 - 2

| / |

3 - 4 - 5 6 - 7

Introduction to classes

To make larger codebases, programmers need to actively improve information hiding. This is the goal of keeping code separated from the rest of the program except through well-defined interfaces.

One method of information hiding is through classes.

A class is a data structure with data (aka members) and functions that act on the data (aka methods).

Any of the data inside of the class can only be operated on through the class method which hides the complexity of the class from the rest of the program.

Large number class 23 April 2013

A large number is an integer that has potential values larger than 4B.

We want an interface that behaves like

largeNum a,b;

cout <<”enter a num”<<endl;

cin>>a;

cin>>b;

cout<<”the sum is”<<a+b<<endl;

Classes can be initialized using constructors. A constructor has the same name as the class and is called automatically by the compiler when the class is created.

A destructor is also called whenever the class is destroyed. If a constructor or destructor is not made explicitly, then the compiler creates one, which does nothing.

The largeNum class will have a vector of integers for its primary data member. One of the methods for the class will be addition, and another will convert from strings to the largeNum.

Complex numbers

A complex number uses i=sqrt(-1) which has the property that i2 = -1.

Any complex number can always be written as

Z=a+ib ; z is a complex number

a and b are real numbers

Normal rules for addition, subtraction and multiplication apply.

Z1=a+ib

Z2=c+id

Z1+z2 = a+ib + c+id

= (a+b)+i(c+d)

Z1z2 = (a+ib)( c+id)

= ac + aid + ibc + i2bd

= (ac – bd) + i(ad + bc)

Z1/z2 = a+ib/ c+id

= a+ib/ c+id \* c-id/c-id

= ac-aid+ibc+bd/c2+d2

= (ac+bd)+i(bc-ad)/ c2+d2

Graphics

With ms windows.

2d 🡪directX

🡪SDL \*simple directory layer

3d 🡪directX

->openGL

Setting up SDL

🡪Makes a new project in codeblocks

🡪Project |build options|

Search directories

Compiler add🡪 ….\SDL-1.2.14\include

Linker add🡪 ....\SDL-1.2.14\lib

Add🡪 …..\SDL-1.2.14\bin

Linker settings

Linker box, right side

-lmingw32 -lSDLmain -lSDL

Copy ciruit.bmp into folder with main.cpp

SDL

* Event driven programming
* External events are the only way to make changes in the program
* Events include mouse/keyboard interactions
* The program must work entirely within 1/60 of a second chunks