CS120 Resources

Ryan Wilson
Ball State University
Department of Computer Science
Honors Thesis
Advisor: Dr. Bonnie McVey
Spring 2000

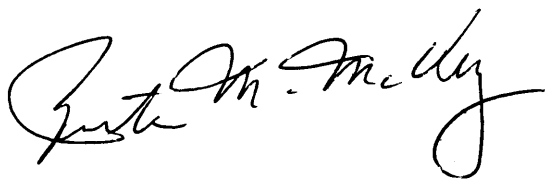

Table of Contents

Section I – CS 120 Resources

Introduction

Computer Science 120 is the first course that every Computer Science major at Ball State University must complete. From the undergraduate catalog, the description is "Introduction to programming using a structured, strongly typed language. Topics include: basic control structures, input/output, procedures and functions, arrays, textfiles, sets, records, introduction to software engineering, and ethics of computing." After reading this description, it is apparent that CS 120 is a rigorous course. Many of the students enrolled in CS120 are first term freshmen that are not accustomed to working and studying at the level required for this course. As a result, a great deal of assistance is needed to help all students succeed. This assistance is provided in several different ways. Dr. McVey and her graduate/undergraduate assistants hold office hours, during which students can get help with homework and programming assignments. Students are also encouraged to contact Dr. McVey and her assistants via email with questions. Appointments with Dr. McVey can be made by students unable to attend any of the available office hours. Finally, Dr. McVey keeps an up-to-date CS120 web site. Despite all that is available, there is still a need for more assistance. CS120 Resources attempts to alleviate this need.

Purpose

CS120 Resources is a web site designed and developed with the intent of providing a resource for students who need assistance in Computer Science 120. The interface to the site is intended to allow students to navigate easily to the information they are seeking. CS120 Resources attempts to provide information about the basic and core concepts taught in CS120. Students must use bsu-cs, the computer science department's Unix system, in order to write and test their programs. Two pages of Unix commands and information are available for students who may have questions about the system. VI is the editor that students must learn and master in order to create their programs. A listing of the most commonly used VI commands is provided on the CS120 Resources site for students. All information available on the site can be printed allowing students to have a hard copy to use when working throughout the semester. A complete listing of keywords used in the C++ programming language can be found with descriptions of the keyword and an example of its use. Also available is a glossary of terms commonly used in the Computer Science field. The glossary is interactive. Students click on the term for which they wish to see a definition, there is a list of terms available, and the definition will appear in a text box on the lower part of the screen. All operators used in the C++ language are listed in order of precedence with other information about the operator and an example of its use. One of the larger parts of the CS120 Resources site is the code examples section. In this area, students will find detailed descriptions of code segments with output results clearly demonstrated. A walkthrough of the common Hello World program is available. Documented segments of code may be downloaded directly to a student's account where the student can compile and execute the code. Executable versions of programs are also available for download. Students wishing to gain a clearer understanding of how loops work in C++ may experiment with the interactive loop examples found in the code examples section. Because CS120 Resources was developed with the intent of being a supplement to Dr. McVey's CS120 web site, there is a direct link to the CS120 web site from the CS120 Resources site. Students not finding the

information they desire on either the CS120 web site or the CS120 Resources web site, or students desiring to learn more, may wish to visit any of the other 28 C++ informational web sites linked directly from the CS120 Resources web site. Finally, students may have a suggestion about improving the CS120 Resources web site. To accommodate these students, there is a suggestion form on the web site where students can enter their ideas and submit them directly to the webmaster.

JavaScript

The purpose of this Honors Thesis was not only to complete a project but also to learn while doing so. As the idea of creating CS120 help site formed, I decided that using JavaScript when programming the web site would not only allow the site to become more interactive and powerful, but it would also require me to learn JavaScript. At the start of this project, I did not know anything about JavaScript. However, I now feel confident in using JavaScript as another programming tool when working with web sites.

JavaScript is programming language developed by Netscape to be used in web pages to allow web sites to be dynamic. HTML alone will always provide the same web site when viewed by different users. JavaScript allows the web site to function differently depending upon the values of variables at the time of execution. JavaScript is not Java, but it can interact with Java on a web site. JavaScript is a client-side programming language. Therefore, the code is executed on the user's computer when a user is viewing a web site containing JavaScript. Ideally, this would not cause any problems as all JavaScript interpreters should interpret the same section of JavaScript code in the same manner. However, this is not the case. Each browser handles JavaScript differently. The two most frequently used browsers today are Internet Explorer and Netscape. When this project began, Netscape was the leading browser statistically in use. Now at the completion of CS120 Resources, Internet Explorer is the leading browser statistically in use. JavaScript code can be written to detect which browser is being used to view the web site and execute code that was written specifically for that browser. This method typically requires twice as much code to be written for one problem, one version of the code for each of the two major browsers, and some overhead to designate which code to run based on the browser detected. Early in the development of CS120 Resources, it was decided that the JavaScript contained on this web site should be developed to work with Netscape. Netscape was picked as the browser to support for two reasons. First, Netscape is the developer of JavaScript and, therefore, is well acquainted with JavaScript. Second, Netscape was the most commonly used browser statistically at the time this decision was made. This decision allowed for more code to be developed for the CS120 Resources web site because time was not spent writing two versions of the same code to accommodate compatibility with Internet Explorer. The opening page of the CS120 Resources web site includes an introduction that explains that this web site has been developed for use with the Netscape browser version 4.0 or later. A link to a free download of the Netscape browser software is provided in this introduction to make it easier for students to browse this site. Internet Explorer has been tested with the CS120 Resources site. Versions 4.0 and 4.1 do work with the site with only a few problems. Version 5.0 seems to work very well with the CS120 Resources site but the Netscape browser still provides the best overall experience. The JavaScript code found in the

CS120 Resources web site is not apparent to the user even though it is running in the background of every page. Clear examples of the JavaScript running can be seen on the interactive loop pages. It is the JavaScript programming in the background that executes the interactive loops based upon the information provided by the user. The change color page also uses JavaScript to change the background color of the web site according to the user's preference. Once the color is changed, JavaScript sets a cookie on the user's machine that will remember the color chosen so that it can be used again on the user's next visit. A cookie is a small file stored on the user's computer that contains a piece of information from a web site. JavaScript is also used to display the definitions of the various computer science terms in the empty text box when a user clicks on a term. There is a portion of JavaScript code running on the CS120 Resources web site that is intended not to ever be executed. This code is executed only if an error occurs in the JavaScript found on the web site. Should an error occur, an error report will be automatically generated containing information about the error and the user will be presented with the option of submitted the error report to the webmaster. Another JavaScript program running in the background calculates the time of day and displays it in the status bar on the lower part of the user's browser. Finally, a small amount of JavaScript code is used to automatically display the most recent update time for each page on each page. This allows the user to know when the page was last modified and is useful for knowing if a bug fix has been implemented or possibly more information has been added to the page. Section II contains all of the JavaScript and HTML code used in created the CS120 Resources web site. A printout of the web page is found before the code that underlies that particular web page.

Future

The CS120 Resources web site has been developed to aid CS120 students as they learn. A large amount of information is available at this site but it is in no way an exhaustive resource. The CS120 Resources web site was designed to make it easy to add more information and pages to it in the future. The web site is based on a frame system that displays a menu of available topics on the left and the current topic on the right side of the screen. New pages developed for the CS120 Resources web site can be linked from added topics to the menu. This web site is intended to grow and change over time to reflect the current needs of CS120. Student input and suggestions will greatly influence the future of this web site. HTML and JavaScript technology continues to develop and improve. As a result, the code used to create and run the CS120 Resources web site will continue to be supported by future web browsers.

Acknowledgements

First, and foremost, I would like to thank my advisor, Dr. Bonnie McVey. Without her help and guidance this project would not have been possible. I'd also like to thank everyone who helped me test the code on the web site including, but not limited to, Charlotte, Jason, Jeff, Julie, Justin, and Rob. Another round of thanks goes to those that helped to generate ideas to be used on this web site. Finally, I'd like to give an early thanks to anyone who uses the CS120 Resources web site in the future and offers suggestions on how to improve it.

Section II – Source Code

CS 120 Resources

Please select a link from the list below.

CS 120 Homepage
Change Color
C++ Keywords
C++ Glossary
C++ Operators
Code Examples
Opening Page
Other Sites
Suggestions
Unix Commands
Unix Info
VI Commands

Please choose your tutuorial from the listing in the left frame. When you click on the link, the page will load in this frame.

Note: this site is viewed best at 800X600 resolution. Also, the JavaScript running on this site was developed for Netscape 4.0, Internet Explorer 5.0, or better. You may experience compatibility problems if you are using older versions or any other browser. You can download the current version of Netscape here free.

```
<html>
<head>
<!-- Author: Ryan Wilson -->
    <title>CS 120 Resources</title>
<script>
// This function displays the time in the status line.
// Invoke it once to activate the clock; it will call itself from then on.
function display_time_in_status_line()
{
    var d = new Date();  // Get current time
    var h = d.getHours(); // Extract hours: 0 to 23
    var m = d.getMinutes(); // Extract minutes: 0 to 59
    var ampm = ( h >= 12)?"PM":"AM"; //Is it am or pm?
    if (h > 12)
        h-=12; //Convert 24-hour format to 12-hour
    if ( h == 0)
        h = 12; // Convert 0 o'clock to midnight
    if (m < 10)
        m = "0" + m;  //Convert 0 minutes to 00 minutes, etc.
    var t = h + ':' + m + ' ' + ampm;  // Put it all together

    defaultStatus = t;   //Display it in the status line.

    //Arrange to do it all again in 1 minute.
    setTimeout("display_time_in_status_line()", 60000); //60000 ms is 1 min.
}
</SCript>
</head>
<frameset cols="22%,*">
          <frame src="left.html" noresize>
          <frame src="right.html" name="fill">
<noframes>Sorry, this document can only be viewed with Netscape 4.0 or
IE 5.0 or later.
</noframes>
</frameset>
</html>
```

```
<html>
<head>
<!-- Author: Ryan Wilson -->
    <title>CS 120 Resources</title>
<script>
// This function displays the time in the status line.
// Invoke it once to activate the clock; it will call itself from then on.
function display_time_in_status_line()
{
    var d = new Date();  // Get current time
    var h = d.getHours(); // Extract hours: 0 to 23
    var m = d.getMinutes(); // Extract minutes: 0 to 59
    var ampm = ( h >= 12)?"PM":"AM"; //Is it am or pm?
    if (h > 12)
        h-=12; //Convert 24-hour format to 12-hour
    if ( h == 0)
        h = 12; // Convert 0 o'clock to midnight
    if (m < 10)
        m = "0" + m;  //Convert 0 minutes to 00 minutes, etc.
    var t = h + ':' + m + ' ' + ampm;  // Put it all together

    defaultStatus = t;   //Display it in the status line.

    //Arrange to do it all again in 1 minute.
    setTimeout("display_time_in_status_line()", 60000); //60000 ms is 1 min.
}

var bcolor;
var allcookies = document.cookie;
var pos = allcookies.indexOf("color=");
if (pos != -1)
{
    var start = pos + 6;
    var end = allcookies.indexOf(";", start);
    if (end == -1)
        end = allcookies.length;
    var bcolor = allcookies.substring(start, end);
}
else
    bcolor = "white";

function setbgcolor(bcolor)
{
    document.bgColor = bcolor;
}
</SCRIPT>

</head>

<body text="#0D0F58" link="#760B10" alink="#544C51"
onLoad="display_time_in_status_line(); setbgcolor(bcolor);">

<h1>CS 120 Resources</h1>
Please select a link from the list below.
<br>
<hr>
<a href="http://www.cs.bsu.edu/homepages/cs120/spring00/index.html" target="_blank">
CS 120 Homepage</a><br>
<a href="changecolor.html" target="fill">Change Color</a><br>
<a href="keywords.html" target="fill">C++ Keywords</a><br>
<a href="definitions.html" target="fill">C++ Glossary</A><br>
<a href="operators.html" target="fill">C++ Operators</a><br>
<a href="leftcode.html" >Code Examples</a><br>
<a href="index.html" target="_top">Opening Page</a><br>
<a href="otherlinks.html" target="fill">Other Sites</a><br>
<a href="suggestion.html" target="fill">Suggestions</a><br>
<a href="unixcmds.htm" target="fill">Unix Commands</a><br>
<a href="unixsys.htm" target="fill">Unix Info</a><br>
<a href="vicmds.htm" target="fill">VI Commands</a><br>
</body>
</html>
```

```
<html>
<head>
<!-- Author: Ryan Wilson -->
<script>

// This function displays the time in the status line.
// Invoke it once to activate the clock; it will call itself from then on.
function display_time_in_status_line()
{
    var d = new Date();  // Get current time
    var h = d.getHours(); // Extract hours: 0 to 23
    var m = d.getMinutes(); // Extract minutes: 0 to 59
    var ampm = ( h >= 12)?"PM":"AM"; //Is it am or pm?
    if (h > 12)
        h-=12; //Convert 24-hour format to 12-hour
    if ( h == 0)
        h = 12; // Convert 0 o'clock to midnight
    if (m < 10)
        m = "0" + m;  //Convert 0 minutes to 00 minutes, etc.
    var t = h + ':' + m + ' ' + ampm;  // Put it all together

    defaultStatus = t;   //Display it in the status line.

    //Arrange to do it all again in 1 minute.
    setTimeout("display_time_in_status_line()", 60000); //60000 ms is 1 min.
}

// A variable we use to ensure that each error window we create is unique.
var error_count = 0;

// Email address to send the report to.
var email = "rawilson@bsuvc.bsu.edu";

// Define the error handler.  It generates an HTML form so the user
// can report the error to the author.
function report_error(msg, url, line)
{
    var w = window.open("","error"+error_count++,"resizable,status,width=625,height=400");
    // arguements... url (none specified), Name (force it to be unique), Features
    var d = w.document; // We use this variable to save typing.

    // Output an HTML document, including a form, into the new window.
    d.write('<DIV align=center>');
    d.write('<FONT SIZE=7 FACE="helvetica"><B>');
    d.write('OOPS.... A JavaScript Error Has Occurred!');
    d.write('</B></FONT><BR><HR SIZE=4 WIDTH="80%">');
    d.write('<FORM ACTION="mailto:' + email + '" METHOD=post');
    d.write(' ENCTYPE="text/plain">');
    d.write('<FONT SIZE=3>');
    d.write('<I>CLick the "Report Error" button to send a bug report.<I><BR>');
    d.write('<INPUT TYPE="submit" VALUE="Report Error">  ');
    d.write('<INPUT TYPE="button" VALUE="Dismiss" onClick="self.close()">');
    d.write('</DIV><DIV align=right>');
    d.write('<BR>Your name <I>(optional)</I>: ');
    d.write('<INPUT SIZE=42 NAME="name" VALUE="">');
    d.write('<BR>Error Message: ');
    d.write('<INPUT SIZE=42 NAME="message" VALUE="' + msg + '">');
    d.write('<BR>Document: <INPUT SIZE=42 NAME="url" VALUE="' + url + '">');
    d.write('<BR>Line Number: <INPUT SIZE=42 NAME="line" VALUE="' + line +'">');
    d.write('<BR>Browser Version: ');
    d.write('<INPUT SIZE=42 NAME="version" VALUE="'+navigator.userAgent + '">');
    d.write('</DIV></FONT>');
    d.write('</FORM>');
    // Remember to close the document when we're done.
    d.close();

    // Return true from this error handler, so that JavaScript does not
    // display it own error dialog.
    return true;
}

// Before the event handler can take effect, we have to register it
// for a particular window.
self.onerror = report_error;
```

```
// The below code sets the background color based on the value found in the
// cookie set by the color select page.
var bcolor;
var allcookies = document.cookie;
var pos = allcookies.indexOf("color=");
if (pos != -1)
{
    var start = pos + 6;
    var end = allcookies.indexOf(";", start);
    if (end == -1)
        end = allcookies.length;
    var bcolor = allcookies.substring(start, end);
}
else
    bcolor = "white";

function setbgcolor(bcolor)
{
    document.bgColor = bcolor;
}
//end color setting code
</SCRIPT>

</head>
<body text="#0D0F58" link="#760B10" alink="#544C51"
onLoad="display_time_in_status_line(); setbgcolor(bcolor);">

<p>
Please choose your tutorial from the listing in the left frame.
When you click on the link, the page will load in this frame.
</p>

<p><b>Note:</b> This site is viewed best at 800X600 resolution.  Also, the
JavaScript running on this site was developed for Netscape 4.0, Internet
Explorer 5.0, or better.  You may experience compatibility problems if you are
using older versions or any other browser.
You can download the current version of Netscape
<a href="http://home.netscape.com/computing/download/index.html?cp=hom06tdow" target
    ="_blank">
here</a> free.
</p>

<hr>
<br>
<p align="center">
<a href="http://www.bsu.edu/home.html" target="_top">
<img src="gifs/home.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="mailto:ASKBSU@bsu.edu" target="_top">
<img src="gifs/ask.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="http://www.bsu.edu/cgi-bin/guest" target="_top">
<img src="gifs/guest.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="http://www.bsu.edu/UP/about/bsu.html" target="_top">
<img src="gifs/about.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="http://www.bsu.edu/UP/map/indiana.html" target="_top">
<img src="gifs/map.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="http://www.bsu.edu/htmls/search.html" target="_top">
<img src="gifs/search.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="http://www.bsu.edu/home_help.html" target="_top">
<img src="gifs/help.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<br>
<font size="-1">
<a href="mailto:rawilson@bsuvc.bsu.edu">Author</a> |
Last Update: <SCRIPT>document.write(document.lastModified);</SCRIPT> |
<a href="mailto:rawilson@bsuvc.bsu.edu">Technical comments to the Webmaster</a>
</font>
<br>
```

```
<font size="-4">Ball State University practices equal opportunity in education
and employment and is strongly and actively committed to diversity within its
community.
</font>
</p>

<p align="center">
<em>
<font size="-4">Links contained in this file to information provided
by organizations other than Ball State University's Computer Science Department
 are presented as a service and neither constitute nor imply endorsement or
 warranty.
</font>
</em>
</p>
</body>
</html>
```

Here you can change the background color of all documents on this site. To change the color, simply click on the name of the color you wish to use in the table below. If you do not find a color that you like, you can manually enter either a color name or an HTML color code in the text box below the table and click the submit button. You will need to reload the left frame in order for color changes to take place in that frame. Once you set your color, the background color of this site will reflect your preference until you change it or for one year, whichever happens first.

Default	Aliceblue	Antiquewhite	Aqua	Aquamarine
azure	beige	bisque	blanchedalmond	blue
blueviolet	brown	burlywood	cadetblue	Chartreuse
Chocolate	Coral	Cornflowerblue	Cornsilk	Crimson
Cyan	darkblue	darkcyan	darkgoldenrod	darkgray
darkgreen	darkkhaki	darkmagenta	darkolivegreen	darkorange
darkorchid	darkred	darksalmon	darkseagreen	darkslateblue
darkslategray	darkturquoise	darkviolet	deeppink	Deepskyblue
dimgray	dodgerblue	firebrick	floralwhite	forestgreen
gainsboro	ghostwhite	gold	goldenrod	gray
green	greenyellow	honeydew	hotpink	indianred
ivory	khaki	lavender	lavenderblush	lawngreen
lemonchiffon	lightblue	lightcoral	lightcyan	lightgoldenrod
lightgoldenrodyellow	lightgray	lightgreen	lightpink	lightsalmon
lightseagreen	lightskyblue	lightslateblue	lightslategray	lightsteelblue
lightyellow	limegreen	linen	magenta	maroon
mediumaquamarine	mediumblue	mediumorchid	mediumpurple	mediumseagreen
mediumslateblue	mediumspringgreen	mediumturquoise	mediumvioletred	midnightblue
mintcream	mistyrose	moccasin	navajowhite	navy
navyblue	oldlace	olivedrab	orange	orangered
orchid	palegoldenrod	palegreen	paleturquoise	palevioletred
papayawhip	peachpuff	peru	pink	plum
powderblue	purple	red	rosybrown	royalblue
saddlebrown	salmon	sandybrown	seagreen	seashell
sienna	skyblue	slateblue	slategray	snow
springgreen	steelblue	tan	thistle	tomato
turquoise	violet	violetred	wheat	white
whitesmoke	yellow	yellowgreen		

If you would like to manually pick your own color then please enter the name of the color in the following box and press the submit button. HTML Hex codes will work but you must put a # before the code in order for the browser to recognize it properly.

submit

```
<html>
<head>
<!-- Author: Ryan Wilson -->
<script>
// This function displays the time in the status line.
// Invoke it once to activate the clock; it will call itself from then on.
function display_time_in_status_line()
{
    var d = new Date();  // Get current time
    var h = d.getHours(); // Extract hours: 0 to 23
    var m = d.getMinutes(); // Extract minutes: 0 to 59
    var ampm = ( h >= 12)?"PM":"AM"; //Is it am or pm?
    if (h > 12)
        h-=12; //Convert 24-hour format to 12-hour
    if ( h == 0)
        h = 12; // Convert 0 o'clock to midnight
    if (m < 10)
        m = "0" + m;  //Convert 0 minutes to 00 minutes, etc.
    var t = h + ':' + m + ' ' + ampm;  // Put it all together

    defaultStatus = t;   //Display it in the status line.

    //Arrange to do it all again in 1 minute.
    setTimeout("display_time_in_status_line()", 60000); //60000 ms is 1 min.
}

// A variable we use to ensure that each error window we create is unique.
var error_count = 0;

// Email address to send the report to.
var email = "rawilson@bsuvc.bsu.edu";

// Define the error handler.  It generates an HTML form so the user
// can report the error to the author.
function report_error(msg, url, line)
{
    var w = window.open("","error"+error_count++,"resizable,status,width=625,height=400");
    // arguements... url (none specified), Name (force it to be unique), Features
    var d = w.document; // We use this variable to save typing.

    // Output an HTML document, including a form, into the new window.
    d.write('<DIV align=center>');
    d.write('<FONT SIZE=7 FACE="helvetica"><B>');
    d.write('OOPS.... A JavaScript Error Has Occurred!');
    d.write('</B></FONT><BR><HR SIZE=4 WIDTH="80%">');
    d.write('<FORM ACTION="mailto:' + email + '" METHOD=post');
    d.write(' ENCTYPE="text/plain">');
    d.write('<FONT SIZE=3>');
    d.write('<I>CLick the "Report Error" button to send a bug report.<I><BR>');
    d.write('<INPUT TYPE="submit" VALUE="Report Error">  ');
    d.write('<INPUT TYPE="button" VALUE="Dismiss" onClick="self.close()">');
    d.write('</DIV><DIV align=right>');
    d.write('<BR>Your name <I>(optional)</I>: ');
    d.write('<INPUT SIZE=42 NAME="name" VALUE="">');
    d.write('<BR>Error Message: ');
    d.write('<INPUT SIZE=42 NAME="message" VALUE="' + msg + '">');
    d.write('<BR>Document: <INPUT SIZE=42 NAME="url" VALUE="' + url + '">');
    d.write('<BR>Line Number: <INPUT SIZE=42 NAME="line" VALUE="' + line +'">');
    d.write('<BR>Browser Version: ');
    d.write('<INPUT SIZE=42 NAME="version" VALUE="'+navigator.userAgent + '">');
    d.write('</DIV></FONT>');
    d.write('</FORM>');
    // Remember to close the document when we're done.
    d.close();

    // Return true from this error handler, so that JavaScript does not
    // display it own error dialog.
    return true;
}

// Before the event handler can take effect, we have to register it
// for a particular window.
self.onerror = report_error;
```

```
var color;
var nextyear = new Date();
nextyear.setFullYear(nextyear.getFullYear() + 1);

function setcookie(color)
{
    document.cookie = "color=" + color + "; expires=" + nextyear.toGMTString();
}

// The code below sets the background color based on a value found in a cookie
// set by the changecolor page.
var bcolor;
var allcookies = document.cookie;
var pos = allcookies.indexOf("color=");
if (pos != -1)
{
    var start = pos + 6;
    var end = allcookies.indexOf(";", start);
    if (end == -1)
        end = allcookies.length;
    var bcolor = allcookies.substring(start, end);
}
else
    bcolor = "white";

function setbgcolor(bcolor)
{
    document.bgColor = bcolor;
}
</SCRIPT>

</head>

<body text="#0D0F58" link="#760B10" alink="#544C51"
onLoad="display_time_in_status_line(); setbgcolor(bcolor);">

<p>
Here you can change the background color of all documents on this site.  To
change the color, simply click on the name of the color you wish to use in the
table below.  If you do not find a color that you like, you can manually enter
either a color name or an HTML color code in the text box below the table and
click the submit button.  You will need to reload the left frame in order for
color changes to take place in that frame.  Once you set your color, the
background color of this site will reflect your preference until you change it
or for one year, whichever happens first.
</p>
<center>
<table border>
<td bgcolor="#F5EEDD"><a href="changecolor.html" onclick="javascript: color='#F5EEDD';
    setcookie(color); parent.frames[0].location.reload(); return true;">Default</a></td>
<td bgcolor=Aliceblue><a href="changecolor.html" onclick="javascript: color='Aliceblue';
    setcookie(color); parent.frames[0].location.reload(); return true;">Aliceblue</a></td>
<td bgcolor=antiquewhite><a href="changecolor.html" onclick="javascript: color
    ='antiquewhite'; setcookie(color); parent.frames[0].location.reload(); return true;">
    Antiquewhite</a></td>
<td bgcolor=Aqua><a href="changecolor.html" onclick="javascript: color='Aqua'; setcookie
    (color); parent.frames[0].location.reload(); return true;">Aqua</a></td>
<td bgcolor=Aquamarine><a href="changecolor.html" onclick="javascript: color='Aquamarine';
    setcookie(color); parent.frames[0].location.reload(); return true;">Aquamarine</a></td>
<tr>
<td bgcolor=azure><a href="changecolor.html" onclick="javascript: color='azure'; setcookie
    (color); parent.frames[0].location.reload(); return true;">azure</a></td>
<td bgcolor=beige><a href="changecolor.html" onclick="javascript: color='beige'; setcookie
    (color); parent.frames[0].location.reload(); return true;">beige</a></td>
<td bgcolor=bisque><a href="changecolor.html" onclick="javascript: color='bisque'; setcookie
    (color); parent.frames[0].location.reload(); return true;">bisque</a></td>
<td bgcolor=blanchedalmond><a href="changecolor.html" onclick="javascript: color
    ='blanchedalmond'; setcookie(color); parent.frames[0].location.reload(); return true;">
    blanchedalmond</a></td>
<td bgcolor=blue><a href="changecolor.html" onclick="javascript: color='blue'; setcookie
    (color); parent.frames[0].location.reload(); return true;">blue</a></td>
<tr>
<td bgcolor=blueviolet><a href="changecolor.html" onclick="javascript: color='blueviolet';
    setcookie(color); parent.frames[0].location.reload(); return true;">blueviolet</a></td>

<td bgcolor=Deepskyblue><a href="changecolor.html" onclick="javascript: color='Deepskyblue';
    setcookie(color); parent.frames[0].location.reload(); return true;">Deepskyblue</a></td>
<Tr>
<td bgcolor=dimgray><a href="changecolor.html" onclick="javascript: color='dimgray';
    setcookie(color); parent.frames[0].location.reload(); return true;">dimgray</a></td>
<td bgcolor=dodgerblue><a href="changecolor.html" onclick="javascript: color='dodgerblue';
```

```
    setcookie(color); parent.frames[0].location.reload(); return true;">dodgerblue</a></td>
<td bgcolor=firebrick><a href="changecolor.html" onclick="javascript: color='firebrick';
    setcookie(color); parent.frames[0].location.reload(); return true;">firebrick</a></td>
<td bgcolor=floralwhite><a href="changecolor.html" onclick="javascript: color='floralwhite';
    setcookie(color); parent.frames[0].location.reload(); return true;">floralwhite</a></td>
<td bgcolor=forestgreen><a href="changecolor.html" onclick="javascript: color='forestgreen';
    setcookie(color); parent.frames[0].location.reload(); return true;">forestgreen</a></td>
<tr>
<td bgcolor=gainsboro><a href="changecolor.html" onclick="javascript: color='gainsboro';
    setcookie(color); parent.frames[0].location.reload(); return true;">gainsboro</a></td>
<td bgcolor=ghostwhite><a href="changecolor.html" onclick="javascript: color='ghostwhite';
    setcookie(color); parent.frames[0].location.reload(); return true;">ghostwhite</a></td>
<td bgcolor=gold><a href="changecolor.html" onclick="javascript: color='gold'; setcookie
    (color); parent.frames[0].location.reload(); return true;">gold</a></td>
<td bgcolor=goldenrod><a href="changecolor.html" onclick="javascript: color='goldenrod';
    setcookie(color); parent.frames[0].location.reload(); return true;">goldenrod</a></td>
<td bgcolor=gray><a href="changecolor.html" onclick="javascript: color='gray'; setcookie
    (color); parent.frames[0].location.reload(); return true;">gray</a></td>
<tr>
<td bgcolor=green><a href="changecolor.html" onclick="javascript: color='green'; setcookie
    (color); parent.frames[0].location.reload(); return true;">green</a></td>
<td bgcolor=greenyellow><a href="changecolor.html" onclick="javascript: color='greenyellow';
    setcookie(color); parent.frames[0].location.reload(); return true;">greenyellow</a></td>
<td bgcolor=honeydew><a href="changecolor.html" onclick="javascript: color='honeydew';
    setcookie(color); parent.frames[0].location.reload(); return true;">honeydew</a></td>
<td bgcolor=hotpink><a href="changecolor.html" onclick="javascript: color='hotpink';
    setcookie(color); parent.frames[0].location.reload(); return true;">hotpink</a></td>
<td bgcolor=indianred><a href="changecolor.html" onclick="javascript: color='indianred';
    setcookie(color); parent.frames[0].location.reload(); return true;">indianred</a></td>
<tr>
<td bgcolor=ivory><a href="changecolor.html" onclick="javascript: color='ivory'; setcookie
    (color); parent.frames[0].location.reload(); return true;">ivory</a></td>
<td bgcolor=khaki><a href="changecolor.html" onclick="javascript: color='khaki'; setcookie
    (color); parent.frames[0].location.reload(); return true;">khaki</a></td>
<td bgcolor=lavender><a href="changecolor.html" onclick="javascript: color='lavender';
    setcookie(color); parent.frames[0].location.reload(); return true;">lavender</a></td>
<td bgcolor=lavenderblush><a href="changecolor.html" onclick="javascript: color
    ='lavenderblush'; setcookie(color); parent.frames[0].location.reload(); return true;">
    lavenderblush</a></td>
<td bgcolor=lawngreen><a href="changecolor.html" onclick="javascript: color='lawngreen';
    setcookie(color); parent.frames[0].location.reload(); return true;">lawngreen</a></td>
<tr>
<td bgcolor=lemonchiffon><a href="changecolor.html" onclick="javascript: color
    ='lemonchiffon'; setcookie(color); parent.frames[0].location.reload(); return true;">
    lemonchiffon</a></td>
<td bgcolor=lightblue><a href="changecolor.html" onclick="javascript: color='lightblue';
    setcookie(color); parent.frames[0].location.reload(); return true;">lightblue</a></td>
<td bgcolor=lightcoral><a href="changecolor.html" onclick="javascript: color='lightcoral';
    setcookie(color); parent.frames[0].location.reload(); return true;">lightcoral</a></td>
<td bgcolor=lightcyan><a href="changecolor.html" onclick="javascript: color='lightcyan';
    setcookie(color); parent.frames[0].location.reload(); return true;">lightcyan</a></td>
<td bgcolor=lightgoldenrod><a href="changecolor.html" onclick="javascript: color
    ='lightgoldenrod'; setcookie(color); parent.frames[0].location.reload(); return true;">
    lightgoldenrod</a></td>
<tr>
<td bgcolor=lightgoldenrodyellow><a href="changecolor.html" onclick="javascript: color
    ='lightgoldenrodyellow'; setcookie(color); parent.frames[0].location.reload(); return
    true;">lightgoldenrodyellow</a></td>
<td bgcolor=lightgray><a href="changecolor.html" onclick="javascript: color='lightgray';
    setcookie(color); parent.frames[0].location.reload(); return true;">lightgray</a></td>
<td bgcolor=lightgreen><a href="changecolor.html" onclick="javascript: color='lightgreen';
    setcookie(color); parent.frames[0].location.reload(); return true;">lightgreen</a></td>
<td bgcolor=lightpink><a href="changecolor.html" onclick="javascript: color='lightpink';
    setcookie(color); parent.frames[0].location.reload(); return true;">lightpink</a></td>
<td bgcolor=lightsalmon><a href="changecolor.html" onclick="javascript: color='lightsalmon';
    setcookie(color); parent.frames[0].location.reload(); return true;">lightsalmon</a></td>
<tr>
<td bgcolor=lightseagreen><a href="changecolor.html" onclick="javascript: color
    ='lightseagreen'; setcookie(color); parent.frames[0].location.reload(); return true;">
    lightseagreen</a></td>
<td bgcolor=lightskyblue><a href="changecolor.html" onclick="javascript: color
    ='lightskyblue'; setcookie(color); parent.frames[0].location.reload(); return true;">
    lightskyblue</a></td>
<td bgcolor=lightslateblue><a href="changecolor.html" onclick="javascript: color
```

```
    ='lightslateblue'; setcookie(color); parent.frames[0].location.reload(); return true;">
    lightslateblue</a></td>
<td bgcolor=lightslategray><a href="changecolor.html" onclick="javascript: color
    ='lightslategray'; setcookie(color); parent.frames[0].location.reload(); return true;">
    lightslategray</a></td>
<td bgcolor=lightsteelblue><a href="changecolor.html" onclick="javascript: color
    ='lightsteelblue'; setcookie(color); parent.frames[0].location.reload(); return true;">
    lightsteelblue</a></td>
<tr>
<td bgcolor=lightyellow><a href="changecolor.html" onclick="javascript: color='lightyellow';
    setcookie(color); parent.frames[0].location.reload(); return true;">lightyellow</a></td>
<td bgcolor=limegreen><a href="changecolor.html" onclick="javascript: color='limegreen';
    setcookie(color); parent.frames[0].location.reload(); return true;">limegreen</a></td>
<td bgcolor=linen><a href="changecolor.html" onclick="javascript: color='linen'; setcookie
    (color); parent.frames[0].location.reload(); return true;">linen</a></td>
<td bgcolor=magenta><a href="changecolor.html" onclick="javascript: color='magenta';
    setcookie(color); parent.frames[0].location.reload(); return true;">magenta</a></td>
<td bgcolor=maroon><a href="changecolor.html" onclick="javascript: color='maroon'; setcookie
    (color); parent.frames[0].location.reload(); return true;">maroon</a></td>
<tr>
<td bgcolor=mediumaquamarine><a href="changecolor.html" onclick="javascript: color
    ='mediumaquamarine'; setcookie(color); parent.frames[0].location.reload(); return true;">
    mediumaquamarine</a></td>
<td bgcolor=mediumblue><a href="changecolor.html" onclick="javascript: color='mediumblue';
    setcookie(color); parent.frames[0].location.reload(); return true;">mediumblue</a></td>
<td bgcolor=mediumorchid><a href="changecolor.html" onclick="javascript: color
    ='mediumorchid'; setcookie(color); parent.frames[0].location.reload(); return true;">
    mediumorchid</a></td>
<td bgcolor=mediumpurple><a href="changecolor.html" onclick="javascript: color
    ='mediumpurple'; setcookie(color); parent.frames[0].location.reload(); return true;">
    mediumpurple</a></td>
<td bgcolor=mediumseagreen><a href="changecolor.html" onclick="javascript: color
    ='mediumseagreen'; setcookie(color); parent.frames[0].location.reload(); return true;">
    mediumseagreen</a></td>
<tr>
<td bgcolor=mediumslateblue><a href="changecolor.html" onclick="javascript: color
    ='mediumslateblue'; setcookie(color); parent.frames[0].location.reload(); return true;">
    mediumslateblue</a></td>
<td bgcolor=mediumspringgreen><a href="changecolor.html" onclick="javascript: color
    ='mediumspringgreen'; setcookie(color); parent.frames[0].location.reload(); return true
    ;">mediumspringgreen</a></td>
<td bgcolor=mediumturquoise><a href="changecolor.html" onclick="javascript: color
    ='mediumturquoise'; setcookie(color); parent.frames[0].location.reload(); return true;">
    mediumturquoise</a></td>
<td bgcolor=mediumvioletred><a href="changecolor.html" onclick="javascript: color
    ='mediumvioletred'; setcookie(color); parent.frames[0].location.reload(); return true;">
    mediumvioletred</a></td>
<td bgcolor=midnightblue><a href="changecolor.html" onclick="javascript: color
    ='midnightblue'; setcookie(color); parent.frames[0].location.reload(); return true;">
    midnightblue</a></td>
<tr>
<td bgcolor=mintcream><a href="changecolor.html" onclick="javascript: color='mintcream';
    setcookie(color); parent.frames[0].location.reload(); return true;">mintcream</a></td>
<td bgcolor=mistyrose><a href="changecolor.html" onclick="javascript: color='mistyrose';
    setcookie(color); parent.frames[0].location.reload(); return true;">mistyrose</a></td>
<td bgcolor=moccasin><a href="changecolor.html" onclick="javascript: color='moccasin';
    setcookie(color); parent.frames[0].location.reload(); return true;">moccasin</a></td>
<td bgcolor=navajowhite><a href="changecolor.html" onclick="javascript: color='navajowhite';
    setcookie(color); parent.frames[0].location.reload(); return true;">navajowhite</a></td>
<td bgcolor=navy><a href="changecolor.html" onclick="javascript: color='navy'; setcookie
    (color); parent.frames[0].location.reload(); return true;">navy</a></td>
<tr>
<td bgcolor=navyblue><a href="changecolor.html" onclick="javascript: color='navyblue';
    setcookie(color); parent.frames[0].location.reload(); return true;">navyblue</a></td>
<td bgcolor=oldlace><a href="changecolor.html" onclick="javascript: color='oldlace';
    setcookie(color); parent.frames[0].location.reload(); return true;">oldlace</a></td>
<td bgcolor=olivedrab><a href="changecolor.html" onclick="javascript: color='olivedrab';
    setcookie(color); parent.frames[0].location.reload(); return true;">olivedrab</a></td>
<td bgcolor=orange><a href="changecolor.html" onclick="javascript: color='orange'; setcookie
    (color); parent.frames[0].location.reload(); return true;">orange</a></td>
<td bgcolor=orangered><a href="changecolor.html" onclick="javascript: color='orangered';
    setcookie(color); parent.frames[0].location.reload(); return true;">orangered</a></td>
<tr>
<td bgcolor=orchid><a href="changecolor.html" onclick="javascript: color='orchid'; setcookie
```

```
    (color); parent.frames[0].location.reload(); return true;">orchid</a></td>
<td bgcolor=palegoldenrod><a href="changecolor.html" onclick="javascript: color
    ='palegoldenrod'; setcookie(color); parent.frames[0].location.reload(); return true;">
    palegoldenrod</a></td>
<td bgcolor=palegreen><a href="changecolor.html" onclick="javascript: color='palegreen';
    setcookie(color); parent.frames[0].location.reload(); return true;">palegreen</a></td>
<td bgcolor=paleturquoise><a href="changecolor.html" onclick="javascript: color
    ='paleturquoise'; setcookie(color); parent.frames[0].location.reload(); return true;">
    paleturquoise</a></td>
<td bgcolor=palevioletred><a href="changecolor.html" onclick="javascript: color
    ='palevioletred'; setcookie(color); parent.frames[0].location.reload(); return true;">
    palevioletred</a></td>
<tr>
<td bgcolor=papayawhip><a href="changecolor.html" onclick="javascript: color='papayawhip';
    setcookie(color); parent.frames[0].location.reload(); return true;">papayawhip</a></td>
<td bgcolor=peachpuff><a href="changecolor.html" onclick="javascript: color='peachpuff';
    setcookie(color); parent.frames[0].location.reload(); return true;">peachpuff</a></td>
<td bgcolor=peru><a href="changecolor.html" onclick="javascript: color='peru'; setcookie
    (color); parent.frames[0].location.reload(); return true;">peru</a></td>
<td bgcolor=pink><a href="changecolor.html" onclick="javascript: color='pink'; setcookie
    (color); parent.frames[0].location.reload(); return true;">pink</a></td>
<td bgcolor=plum><a href="changecolor.html" onclick="javascript: color='plum'; setcookie
    (color); parent.frames[0].location.reload(); return true;">plum</a></td>
<tr>
<td bgcolor=powderblue><a href="changecolor.html" onclick="javascript: color='powderblue';
    setcookie(color); parent.frames[0].location.reload(); return true;">powderblue</a></td>
<td bgcolor=purple><a href="changecolor.html" onclick="javascript: color='purple'; setcookie
    (color); parent.frames[0].location.reload(); return true;">purple</a></td>
<td bgcolor=red><a href="changecolor.html" onclick="javascript: color='red'; setcookie
    (color); parent.frames[0].location.reload(); return true;">red</a></td>
<td bgcolor=rosybrown><a href="changecolor.html" onclick="javascript: color='rosybrown';
    setcookie(color); parent.frames[0].location.reload(); return true;">rosybrown</a></td>
<td bgcolor=royalblue><a href="changecolor.html" onclick="javascript: color='royalblue';
    setcookie(color); parent.frames[0].location.reload(); return true;">royalblue</a></td>
<tr>
<td bgcolor=saddlebrown><a href="changecolor.html" onclick="javascript: color='saddlebrown';
    setcookie(color); parent.frames[0].location.reload(); return true;">saddlebrown</a></td>
<td bgcolor=salmon><a href="changecolor.html" onclick="javascript: color='salmon'; setcookie
    (color); parent.frames[0].location.reload(); return true;">salmon</a></td>
<td bgcolor=sandybrown><a href="changecolor.html" onclick="javascript: color='sandybrown';
    setcookie(color); parent.frames[0].location.reload(); return true;">sandybrown</a></td>
<td bgcolor=seagreen><a href="changecolor.html" onclick="javascript: color='seagreen';
    setcookie(color); parent.frames[0].location.reload(); return true;">seagreen</a></td>
<td bgcolor=seashell><a href="changecolor.html" onclick="javascript: color='seashell';
    setcookie(color); parent.frames[0].location.reload(); return true;">seashell</a></td>
<tr>
<td bgcolor=sienna><a href="changecolor.html" onclick="javascript: color='sienna'; setcookie
    (color); parent.frames[0].location.reload(); return true;">sienna</a></td>
<td bgcolor=skyblue><a href="changecolor.html" onclick="javascript: color='skyblue';
    setcookie(color); parent.frames[0].location.reload(); return true;">skyblue</a></td>
<td bgcolor=slateblue><a href="changecolor.html" onclick="javascript: color='slateblue';
    setcookie(color); parent.frames[0].location.reload(); return true;">slateblue</a></td>
<td bgcolor=slategray><a href="changecolor.html" onclick="javascript: color='slategray';
    setcookie(color); parent.frames[0].location.reload(); return true;">slategray</a></td>
<td bgcolor=snow><a href="changecolor.html" onclick="javascript: color='snow'; setcookie
    (color); parent.frames[0].location.reload(); return true;">snow</a></td>
<tr>
<td bgcolor=springgreen><a href="changecolor.html" onclick="javascript: color='springgreen';
    setcookie(color); parent.frames[0].location.reload(); return true;">springgreen</a></td>
<td bgcolor=steelblue><a href="changecolor.html" onclick="javascript: color='steelblue';
    setcookie(color); parent.frames[0].location.reload(); return true;">steelblue</a></td>
<td bgcolor=tan><a href="changecolor.html" onclick="javascript: color='tan'; setcookie
    (color); parent.frames[0].location.reload(); return true;">tan</a></td>
<td bgcolor=thistle><a href="changecolor.html" onclick="javascript: color='thistle';
    setcookie(color); parent.frames[0].location.reload(); return true;">thistle</a></td>
<td bgcolor=tomato><a href="changecolor.html" onclick="javascript: color='tomato'; setcookie
    (color); parent.frames[0].location.reload(); return true;">tomato</a></td>
<tr>
<td bgcolor=turquoise><a href="changecolor.html" onclick="javascript: color='turquoise';
    setcookie(color); parent.frames[0].location.reload(); return true;">turquoise</a></td>
<td bgcolor=violet><a href="changecolor.html" onclick="javascript: color='violet'; setcookie
    (color); parent.frames[0].location.reload(); return true;">violet</a></td>
<td bgcolor=violetred><a href="changecolor.html" onclick="javascript: color='violetred';
    setcookie(color); parent.frames[0].location.reload(); return true;">violetred</a></td>
```

```
<td bgcolor=wheat><a href="changecolor.html" onclick="javascript: color='wheat'; setcookie
    (color); parent.frames[0].location.reload(); return true;">wheat</a></td>
<td bgcolor=white><a href="changecolor.html" onclick="javascript: color='white'; setcookie
    (color); parent.frames[0].location.reload(); return true;">white</a></td>
<tr>
<td bgcolor=whitesmoke><a href="changecolor.html" onclick="javascript: color='whitesmoke';
    setcookie(color); parent.frames[0].location.reload(); return true;">whitesmoke</a></td>
<td bgcolor=yellow><a href="changecolor.html" onclick="javascript: color='yellow'; setcookie
    (color); parent.frames[0].location.reload(); return true;">yellow</a></td>
<td bgcolor=yellowgreen><a href="changecolor.html" onclick="javascript: color='yellowgreen';
    setcookie(color); parent.frames[0].location.reload(); return true;">yellowgreen</a></td>
<tr>
</table>
</center>

<p>
If you would like to manually pick your own color then please enter the name
of the color in the following box and press the submit button.  HTML Hex codes
will work but you must put a # before the code in order for the browser to recognize it
properly.
</p>

<form name="colorp" action="changecolor.html">
<center>
<TEXTAREA NAME="colorpick" value="" ROWS=1 Cols=20 wrap=on></TEXTAREA>
<br>
<input type="submit" value="submit" name="submit1" onclick="setcookie(document.colorp.
    colorpick.value); parent.frames[0].location.reload();" >
</center>
<br>
</FORM>

<hr>
<br>
<p align="center">
<a href="http://www.bsu.edu/home.html" target="_top">
<img src="gifs/home.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="mailto:ASKBSU@bsu.edu" target="_top">
<img src="gifs/ask.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="http://www.bsu.edu/cgi-bin/guest" target="_top">
<img src="gifs/guest.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="http://www.bsu.edu/UP/about/bsu.html" target="_top">
<img src="gifs/about.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="http://www.bsu.edu/UP/map/indiana.html" target="_top">
<img src="gifs/map.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="http://www.bsu.edu/htmls/search.html" target="_top">
<img src="gifs/search.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="http://www.bsu.edu/home_help.html" target="_top">
<img src="gifs/help.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<br>
<font size="-1">
<a href="mailto:rawilson@bsuvc.bsu.edu">Author</a> |
Last Update: <SCRIPT>document.write(document.lastModified);</SCRIPT> |
<a href="mailto:rawilson@bsuvc.bsu.edu">Technical comments to the Webmaster</a>
</font>
<br>
<font size="-4">Ball State University practices equal opportunity in education
and employment and is strongly and actively committed to diversity within its
community.
</font>
</p>

<p align="center">
<em>
<font size="-4">Links contained in this file to information provided
by organizations other than Ball State University's Computer Science Department
 are presented as a service and neither constitute nor imply endorsement or
```

```
warranty.
</font>
</em>
</p>
</body>
</html>
```

C++ Keywords

Keyword	Description	Example
asm	Allows information to be passed to the assembler directly.	asm ("check");
auto	Storage class for objects that exist only within their own block	auto int n;
break	Terminates a loop or a switch statement.	break;
case	Used in a switch statement to specify control expression.	switch (n/10)
catch	Specifies actions to take when an exception occurs.	catch(error)
char	An integer type	char c;
class	Specifies a class declaration	class X { ... };
const	Specifies a constant definition	const int s = 32;
continue	Jumps to beginning of next iteration in a loop	continue;
default	The "otherwise" case in a switch statement	default: sum = 0;
delete	Deallocates memory allocated by a new statement	delete a;
do	Specifies a do..while loop	do {...} while (...);
double	A real number type	double x;
else	Specifies alternative in an if statement	else n = 0;
enum	Used to declare an enumeration type	enum bool { ... };
extern	Storage class for objects declared outside the local block	extern int max;
float	A real number type	float x;
for	Specifies a for loop	for (; ;) ...
friend	Specifies a friend function in a class	friend int f();
goto	Causes execution to jump to a labeled statement	goto error;
if	Specifies an if statement	if (n > 0) ...
inline	Declares a function whose text is to be substituted for its call	inline int f();
int	An integer type	int n;
long	Used to define integer and real types	long double x;
new	Allocates memory	int* p = new int;
operator	Used to declare an operator overload	X operator++();
private	Specifies private declarations in a class	private: int n;
protected	Specifies protected declarations in a class	protected: int n;
public	Specifies public declarations in a class	public: int n;
register	Storage class specifier for objects stored in registers	register int i;
return	Statement that terminates a function and returns a value	return 0;
short	An integer type	short n;
signed	Used to define integer types	signed char c;

sizeof	Operator that returns the number of bytes used to store an object	n = sizeof(float);
static	Storage class of objects that exist for the duration of the program	static int n;
struct	Specifies a structure definition	struct X { ... };
switch	Specifies a switch statement	switch (n) { ... }
template	Specifies a template class	template < class T>
this	Pointer that points to the current object	return *this;
throw	Used to generate an exception	throw X();
try	Specifies a block that contains exception handlers	try { ... }
typedef	Declares a synonym for an existing type	typedef int Num;
union	Specifies a structure whose elements occupy the same storage	union z { ... };
unsigned	Used to define integer types	unsigned int b;
virtual	Declares a member function that is defined in a subclass	virtual int f();
void	Designates the absence of a type	void f();
volatile	Declares objects that can be modified outside of program control	int volatile n;
while	Specifies a while loop	while (n > 0) ...

```
<html>
<head>
<!-- Author: Ryan Wilson -->
<title>C++ Keywords</title>
<script>
// This function displays the time in the status line.
// Invoke it once to activate the clock; it will call itself from then on.
function display_time_in_status_line()
{
    var d = new Date();  // Get current time
    var h = d.getHours(); // Extract hours: 0 to 23
    var m = d.getMinutes(); // Extract minutes: 0 to 59
    var ampm = ( h >= 12)?"PM":"AM"; //Is it am or pm?
    if (h > 12)
        h-=12; //Convert 24-hour format to 12-hour
    if ( h == 0)
        h = 12; // Convert 0 o'clock to midnight
    if (m < 10)
        m = "0" + m;  //Convert 0 minutes to 00 minutes, etc.
    var t = h + ':' + m + ' ' + ampm;  // Put it all together

    defaultStatus = t;   //Display it in the status line.

    //Arrange to do it all again in 1 minute.
    setTimeout("display_time_in_status_line()", 60000); //60000 ms is 1 min.
}

// A variable we use to ensure that each error window we create is unique.
var error_count = 0;

// Email address to send the report to.
var email = "rawilson@bsuvc.bsu.edu";

// Define the error handler.  It generates an HTML form so the user
// can report the error to the author.
function report_error(msg, url, line)
{
    var w = window.open("","error"+error_count++,"resizable,status,width=625,height=400");
    // arguements... url (none specified), Name (force it to be unique), Features
    var d = w.document; // We use this variable to save typing.

    // Output an HTML document, including a form, into the new window.
    d.write('<DIV align=center>');
    d.write('<FONT SIZE=7 FACE="helvetica"><B>');
    d.write('OOPS.... A JavaScript Error Has Occurred!');
    d.write('</B></FONT><BR><HR SIZE=4 WIDTH="80%">');
    d.write('<FORM ACTION="mailto:' + email + '" METHOD=post');
    d.write(' ENCTYPE="text/plain">');
    d.write('<FONT SIZE=3>');
    d.write('<I>CLick the "Report Error" button to send a bug report.<I><BR>');
    d.write('<INPUT TYPE="submit" VALUE="Report Error">  ');
    d.write('<INPUT TYPE="button" VALUE="Dismiss" onClick="self.close()">');
    d.write('</DIV><DIV align=right>');
    d.write('<BR>Your name <I>(optional)</I>: ');
    d.write('<INPUT SIZE=42 NAME="name" VALUE="">');
    d.write('<BR>Error Message: ');
    d.write('<INPUT SIZE=42 NAME="message" VALUE="' + msg + '">');
    d.write('<BR>Document: <INPUT SIZE=42 NAME="url" VALUE="' + url + '">');
    d.write('<BR>Line Number: <INPUT SIZE=42 NAME="line" VALUE="' + line +'">');
    d.write('<BR>Browser Version: ');
    d.write('<INPUT SIZE=42 NAME="version" VALUE="'+navigator.userAgent + '">');
    d.write('</DIV></FONT>');
    d.write('</FORM>');
    // Remember to close the document when we're done.
    d.close();

    // Return true from this error handler, so that JavaScript does not
    // display it own error dialog.
    return true;
}

// Before the event handler can take effect, we have to register it
// for a particular window.
self.onerror = report_error;
```

```
var bcolor;
var allcookies = document.cookie;
var pos = allcookies.indexOf("color=");
if (pos != -1)
{
    var start = pos + 6;
    var end = allcookies.indexOf(";", start);
    if (end == -1)
        end = allcookies.length;
    var bcolor = allcookies.substring(start, end);
}
else
    bcolor = "white";

function setbgcolor(bcolor)
{
    document.bgColor = bcolor;
}
</SCRIPT>

</head>

<body text="#0D0F58" link="#760B10" alink="#544C51"
onLoad="display_time_in_status_line(); setbgcolor(bcolor);">

<h2><center>C++ Keywords</center></h2><BR>

<table border>
<th align=center>Keyword</th>
<th align=center>Description</th>
<th align=center>Example</th>
<tr>
<td>asm</td>
<tD>Allows information to be passed to the assembler directly.</td>
<td>asm ("check");</tD>
<tr>
<td>auto</tD>
<tD>Storage class for objects that exist only within their own block</td>
<td>auto int n;</td>
<tr>
<td>break</tD>
<td>Terminates a loop or a switch statement.</tD>
<td>break;</tD>
<tR>
<td>case</tD>
<tD>Used in a switch statement to specify control expression.</td>
<td>switch (n/10)</td>
<tr>
<td>catch</tD>
<td>Specifies actions to take when an exception occurs.</td>
<td>catch(error)</tD>
<tR>
<td>char</tD>
<td>An integer type</tD>
<td>char c;</tD>
<tr>
<td>class</tD>
<td>Specifies a class declaration</td>
<td>class X { ... };</tD>
<tr>
<td>const</tD>
<td>Specifies a constant definition</td>
<td>const int s = 32;</td>
<tR>
<Td>continue</tD>
<td>Jumps to beginning of next iteration in a loop</tD>
<td>continue;</td>
<tr>
<Td>default</tD>
<tD>The "otherwise" case in a switch statement</tD>
<tD>default: sum = 0;</td>
<tr>
<td>delete</tD>
```

```
<tD>Deallocates memory allocated by a new statement</tD>
<td>delete a;</td>
<tr>
<td>do</tD>
<tD>Specifies a do..while loop</tD>
<td>do {...} while (...);</tD>
<TR>
<tD>double</td>
<tD>A real number type</tD>
<tD>double x;</tD>
<Tr>
<tD>else</tD>
<tD>Specifies alternative in an if statement</tD>
<tD>else n = 0;</tD>
<Tr>
<tD>enum</tD>
<tD>Used to declare an enumeration type</tD>
<TD>enum bool { ... };</tD>
<tr>
<td>extern</tD>
<tD>Storage class for objects declared outside the local block</tD>
<TD>extern int max;</tD>
<tR>
<TD>float</tD>
<tD>A real number type</tD>
<td>float x;</tD>
<Tr>
<td>for</tD>
<tD>Specifies a for loop</tD>
<tD>for ( ; ; ) ...</tD>
<TR>
<td>friend</tD>
<TD>Specifies a friend function in a class</tD>
<Td>friend int f();</tD>
<tr>
<tD>goto</tD>
<TD>Causes execution to jump to a labeled statement</tD>
<tD>goto error;</tD>
<tr>
<tD>if</tD>
<tD>Specifies an if statement</tD>
<td>if (n > 0) ...</tD>
<tr>
<tD>inline</tD>
<tD>Declares a function whose text is to be substituted for its call</tD>
<td>inline int f();</tD>
<tr>
<td>int</td>
<td>An integer type</td>
<TD>int n;</td>
<Tr>
<TD>long</tD>
<tD>Used to define integer and real types</tD>
<Td>long double x;</tD>
<TR>
<tD>new</tD>
<tD>Allocates memory</tD>
<tD>int* p = new int;</tD>
<tr>
<tD>operator</td>
<tD>Used to declare an operator overload</tD>
<Td>X operator++();</tD>
<tR>
<tD>private</tD>
<TD>Specifies private declarations in a class</tD>
<tD>private: int n;</tD>
<tr>
<td>protected</tD>
<TD>Specifies protected declarations in a class</tD>
<td>protected: int n;</tD>
<tR>
<Td>public</tD>
<tD>Specifies public declarations in a class</tD>
<tD>public: int n;</tD>
```

```
<tR>
<TD>register</tD>
<tD>Storage class specifier for objects stored in registers</tD>
<tD>register int i;</tD>
<tR>
<tD>return</tD>
<tD>Statement that terminates a function and returns a value</tD>
<TD>return 0;</tD>
<TR>
<td>short</tD>
<tD>An integer type</tD>
<tD>short n;</tD>
<TR>
<tD>signed</tD>
<td>Used to define integer types</tD>
<TD>signed char c;</tD>
<tr>
<TD>sizeof</tD>
<tD>Operator that returns the number of bytes used to store an object</tD>
<Td>n = sizeof(float);</tD>
<tR>
<TD>static</tD>
<Td>Storage class of objects that exist for the duration of the program</tD>
<TD>static int n;</tD>
<tr>
<TD>struct</tD>
<td>Specifies a structure definition</tD>
<tD>struct X { ... };</tD>
<tr>
<tD>switch</tD>
<tD>Specifies a switch statement</tD>
<Td>switch (n) { ... }</tD>
<tR>
<tD>template</td>
<Td>Specifies a template class</tD>
<TD>template &lt class T&gt</tD>
<tr>
<td>this</tD>
<TD>Pointer that points to the current object</tD>
<TD>return *this;</td>
<TR>
<Td>throw</tD>
<TD>Used to generate an exception</tD>
<td>throw X();</tD>
<TR>
<tD>try</tD>
<TD>Specifies a block that contains exception handlers</tD>
<td>try { ... }</tD>
<TR>
<TD>typedef</tD>
<tD>Declares a synonym for an existing type</tD>
<TD>typedef int Num;</tD>
<TR>
<tD>union</tD>
<TD>Specifies a structure whose elements occupy the same storage</tD>
<tD>union z { ... };</tD>
<TR>
<TD>unsigned</tD>
<TD>Used to define integer types</tD>
<TD>unsigned int b;</tD>
<tR>
<Td>virtual</tD>
<tD>Declares a member function that is defined in a subclass</tD>
<Td>virtual int f();</tD>
<tR>
<TD>void</tD>
<TD>Designates the absence of a type</tD>
<TD>void f();</tD>
<TR>
<Td>volatile</tD>
<tD>Declares objects that can be modified outside of program control</tD>
<Td>int volatile n;</tD>
<TR>
<Td>while</tD>
```

```
<tD>Specifies a while loop</tD>
<tD>while (n > 0) ...</tD>
<tr>
</table>

<hr>
<br>
<p align="center">
<a href="http://www.bsu.edu/home.html" target="_top">
<img src="gifs/home.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="mailto:ASKBSU@bsu.edu" target="_top">
<img src="gifs/ask.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="http://www.bsu.edu/cgi-bin/guest" target="_top">
<img src="gifs/guest.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="http://www.bsu.edu/UP/about/bsu.html" target="_top">
<img src="gifs/about.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="http://www.bsu.edu/UP/map/indiana.html" target="_top">
<img src="gifs/map.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="http://www.bsu.edu/htmls/search.html" target="_top">
<img src="gifs/search.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="http://www.bsu.edu/home_help.html" target="_top">
<img src="gifs/help.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<br>
<font size="-1">
<a href="mailto:rawilson@bsuvc.bsu.edu">Author</a> |
Last Update: <SCRIPT>document.write(document.lastModified);</SCRIPT> |
<a href="mailto:rawilson@bsuvc.bsu.edu">Technical comments to the Webmaster</a>
</font>
<br>
<font size="-4">Ball State University practices equal opportunity in education
and employment and is strongly and actively committed to diversity within its
community.
</font>
</p>

<p align="center">
<em>
<font size="-4">Links contained in this file to information provided
by organizations other than Ball State University's Computer Science Department
are presented as a service and neither constitute nor imply endorsement or
warranty.
</font>
</em>
</p>
</body>
</html>
```

C++ Glossary

Please click your left mouse button on the word that you would like defined. Its definition will appear in the text box at the bottom of the page.

Abstract data type (ADT)	Access function	Actual argument	Algorithm	Array
Base class	Block scope	Calling program	Class (abstract level)	Class (implementation level)
Components	Compound statement	Constructor	Control structures	Data Type
Declaration	Default parameter	Definition	Delete operator	Derived class
Destructor	Dynamic binding	Encapsulation	Enumerated data types	Field
FIFO	File	File scope	File stream	File stream buffer
File window	Fixed repetition loop	Formal parameter	Function	Function header
Function prototype	Global variable	Identifier	Information hiding	Implementation
In-line function	Indirection	Inheritance	Instance	IS-A
Iteration	LIFO	Linked List	List	Local variable
Member	Message	Method	Microprocessor	Multiple inheritance
Nested looping	Nested structure	new Operator	Object	Object-oriented programming
Object program	Overloaded constructor	Parameter	Pointer	Polymorphism
Posttest loops	Pretest loops	Primitive state	Project file (.prj)	Protected member
Pseudocode	Public base class	Queue	Reading	Recursion
Recursive function	Reference parameters	Run-time error	Scope	Selection
Sequence	Single inheritance	Source program	Stack	Standard function
Star,*	Static binding	Structure	Structured design	Structured programming
Syntax error	Value parameters	Virtual function	Whitespace	Writing

---

```
<HTML>
<HEAD>
<!-- Author: Ryan Wilson -->
</HEAD>
<FRAMESET rows="70%,30%">
<FRAME NAME="top" SRC="topdef.html" noresize>
<FRAME NAME="bottom" SRC="bottomdef.html" scrolling=no noresize>
</FRAMESET>
</HTML>
```

```
<html>
<head>
<!-- Author: Ryan Wilson -->
<title>C++ Glossary</title>

<SCRIPT>
// This function displays the time in the status line.
// Invoke it once to activate the clock; it will call itself from then on.
function display_time_in_status_line()
{
    var d = new Date();  // Get current time
    var h = d.getHours(); // Extract hours: 0 to 23
    var m = d.getMinutes(); // Extract minutes: 0 to 59
    var ampm = ( h >= 12)?"PM":"AM"; //Is it am or pm?
    if (h > 12)
        h-=12; //Convert 24-hour format to 12-hour
    if ( h == 0)
        h = 12; // Convert 0 o'clock to midnight
    if (m < 10)
        m = "0" + m;  //Convert 0 minutes to 00 minutes, etc.
    var t = h + ':' + m + ' ' + ampm;  // Put it all together

    defaultStatus = t;   //Display it in the status line.

    //Arrange to do it all again in 1 minute.
    setTimeout("display_time_in_status_line()", 60000); //60000 ms is 1 min.
}

var bcolor;
var allcookies = document.cookie;
var pos = allcookies.indexOf("color=");
if (pos != -1)
{
    var start = pos + 6;
    var end = allcookies.indexOf(";", start);
    if (end == -1)
        end = allcookies.length;
    var bcolor = allcookies.substring(start, end);
}
else
    bcolor = "white";

function setbgcolor(bcolor)
{
    document.bgColor = bcolor;
}
</SCRIPT>

</head>

<body text="#0D0F58" link="#760B10" alink="#544C51"
onLoad="display_time_in_status_line(); setbgcolor(bcolor);">

<h2>
<center>
<b>C++ Glossary</b>
</center>
</h2>

<p>
Please click your left mouse button on the word that you would like defined.
Its definition will appear in the text box at the bottom of the page.
</p>

<p>
<table border="0" cols="5">
<tr>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='A structure that includes both data and related operations with a means to      ↙
    encapsulate the structure details; whereby the structure data is completely hidden from ↙
    its surroundings and the structure operations provide loose coupling of the structure to ↙
    the outside world via a function interface.'; return false;">Abstract data type (ADT)</a ↙
    ></td>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
```

```
    value='A function that only returns the values of the private members of an object.';
    return false;">Access function</a></td>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
    value='A value passed to a function during a function call.'; return false;">Actual
    argument</a></td>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
    value='A series of step-by-step instructions that produce a solution to a problem.';
    return false;">Algorithm</a></td>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
    value='An indexed data structure which is used to store data elements of the same data
    type.'; return false;">Array</a></td>
<tr>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
    value='The parent class of a derived class.'; return false;">Base class</a></td>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
    value='The accessibility, or visibility, of a local variable defined in a given block of
    code, such as a function.'; return false;">Block scope</a></td>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
    value='The program that calls, or invokes, a function.'; return false;">Calling program</
    a></td>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
    value='An interface that defines the behavior of its objects.'; return false;">Class
    (abstract level)</a></td>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
    value='A syntactical unit that describes a set of data and related operations that are
    common to its objects.'; return false;">Class (implementation level)</a></td>
<tr>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
    value='Data elements in a file form components.'; return false;">Components</a></td>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
    value='Several statements framed by curly braces.'; return false;">Compound statement</a
    ></td>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
    value='A special class function that is used to initialize the data members of an object
    automatically when the object is defined'; return false;">Constructor</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
    value='A pattern for controlling the flow of a program module'; return false;">Control
    structures</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
    value='A set of data elements that more or less belong with each other. Examples include
    int, char, float, enum'; return false;">Data Type</a><bR>
<tr>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
    value='A declaration specifies the name and attributes of a value, but does not reserve
    storage. We declare constants'; return false;">Declaration</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
    value='A function parameter that is assigned a default value in the function prototype or
    the funtion header, but not both'; return false;">Default parameter</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
    value='A definition specifies the name and attributes of a variable and also reserves
    storage. Variables are defined'; return false;">Definition</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
    value='The delete operator is used to dellocate memory created dynamically by the new
    operator'; return false;">Delete operator</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
    value='An inherited, or child, class that will include its own members and also include
    members inherited from the base class'; return false;">Derived class</a><bR>
<tr>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
    value='The counterpart of a constructor that is used to clean up an object after it is no
    longer needed. Normally used to deallocate memory allocated to an object by the object
    constructor'; return false;">Destructor</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
    value='Dynamic binding occurs when a polymorphic function is defined for several classes
    in a family but the actual code for the function is not attached, or bound, until
    execution time. A polymorphic function which is dynamically bound is called a virtual
    function'; return false;">Dynamic binding</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
    value='To package data and/or operations into a single well-defined programming unit';
    return false;">Encapsulation</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
    value='A set of data elements that the programmer defines for a particular application';
    return false;">Enumerated data types</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
```

```
    value='An item of meaningful data'; return false;">Field</a><bR>
<tr>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='First-in, first-out; FIFO is associated with queues'; return false;">FIFO</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='A data structure that consists of a sequence of components of the same data type, ↙
    usually associated with program I/O. A means by which the program communciates with the  ↙
    outside world'; return false;">File</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='The scope of a global variable defined prior to main() that is accessible to any  ↙
    block in the same file'; return false;">File scope</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='A file stream provides a channel for data to flow between the program and the      ↙
    outside world'; return false;">File stream</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='The link between a program and the file components'; return false;">File stream   ↙
    buffer</a><bR>
<tr>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='The means for a program to communicate with a file. The file window locates the   ↙
    components within the file for processing'; return false;">File window</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='A loop that will be executed a predetermined number of times'; return false;">    ↙
    Fixed repetition loop</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='A variable used in the function header that receives the value of the respective  ↙
    actual argument in the function call'; return false;">Formal parameter</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='A subprogram that returns a single value, a set of values, or performs some       ↙
    specific task such as I/O'; return false;">Function</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='A statement that forms a common boundary, or interface, between the function and  ↙
    its calling program'; return false;">Function header</a><bR>
<tr>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='A model of the interface to the function that is used by the compiler to check    ↙
    calls to the function for the proper number of arguments and the correct data types of   ↙
    the arguments'; return false;">Function prototype</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='A variable, defined prior to main(), that can be used by all functions of a given ↙
    program, including main(). Global variables have file scope'; return false;">Global      ↙
    variable</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='A unique name associated with a constant, variable, function, data structure,     ↙
    class, object, etc.'; return false;">Identifier</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='Information hiding is accomplished when there exists a binding relationship       ↙
    between the information, or data, and its related operations such that operations outside↙
    of an encapsulated unit can not affect the information inside the unit'; return false;"> ↙
    Information hiding</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='The definition of the function for a class that includes the function header and  ↙
    the body of the function'; return false;">Implementation</a><bR>
<tr>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='A function whose implementation is coded within curly braces, following the      ↙
    function, prototype'; return false;">In-line function</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='Indirection has to do with the levels of addressing it takes to access data';     ↙
    return false;">Indirection</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='That property of object-oriented programming that allows one clas, called a       ↙
    derived class, to share the structure and behavior of another class, called a base       ↙
    class'; return false;">Inheritance</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='In object-oriented programming, an example, or specimen, of a class. We say that  ↙
    an object is an instance of a class'; return false;">Instance</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='The link between a derived class and its base class'; return false;">IS-A</a><bR>
<tr>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='a control structure, also called looping, that causes the program flow to repeat a↙
    finite number of times'; return false;">Iteration</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
```

```
    value='Last-in, first-out, LIFO is associated with stacks'; return false;">LIFO</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='A sequential data structure where, given any element in the list the location of  ↙
    its successor element is specified by an explicit link, rather than by its natural      ↙
    position in the structure'; return false;">Linked List</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='A sequence of data elements whose basic operations are insertion and deletion of  ↙
    elements to and from the list'; return false;">List</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='A variable that is defined within a given block of code, such as a function. Local↙
    variables have block scope'; return false;">Local variable</a><bR>
<tr>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='Any item declared in a structure or class'; return false;">Member</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='A call to a member function'; return false;">Message</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='A function that is a part of a structure or class'; return false;">Method</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='A single integrated circuit (IC) chip that contains the entire central processing ↙
    unti (CPU)'; return false;">Microprocessor</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='Multiple inheritance occurs when the inherited class members can be traced back to↙
    more than one parent class'; return false;">Multiple inheritance</a><bR>
<tr>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='Looping structures that are located within other looping structures'; return false↙
    ;">Nested looping</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='A structure within a structure'; return false;">Nested structure</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='The new operator is used to dynamically allocate memory'; return false;">new      ↙
    Operator</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='An instance, or specimen, of a given class. An object of a given class has the    ↙
    structure and behavior defined by the class which is common to all objects of the same   ↙
    class'; return false;">Object</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='A form of programming whereby data and related operations are specified as classes↙
    whose instances are objects. The data and related operations are so tightly bound such   ↙
    that only those operations defined for a class can affect the class data. This idea of   ↙
    encapsulation and information hidin allows the easy formation of ADTs'; return false;">  ↙
    Object-oriented programming</a><bR>
<tr>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='The binary machine language program generated by a compiler and usually has a file↙
    extension of .obj'; return false;">Object program</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='A constructor that performs different tasks depending on the number and/or type of↙
    arguments that it receives. An overloaded constructor is statically bound at compile     ↙
    time'; return false;">Overloaded constructor</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='A data item that is received by a function in order for it to perform its         ↙
    designated operation'; return false;">Parameter</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='A pointer is used to represent actual machine addreses'; return false;">Pointer</a↙
    ><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='Polymorphism occurs when functions or objects have the same name for different    ↙
    classes of the same family but behave differently'; return false;">Polymorphism</a><bR>
<tr>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='Testing a condition after each loop execution as in the do/while loop structure'; ↙
    return false;">Posttest loops</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='Testing a condition each time before a loop is executed as in the while and for   ↙
    loop structures'; return false;">Pretest loops</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='A known condition that terminates a recursive function call'; return false;">     ↙
    Primitive state</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='A file that identifies the files that need to be compiled and linked to create a  ↙
    given executable program'; return false;">Project file (.prj)</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
```

```
    value='A member of a class that is accessible to both the base class and any derived
    classes of the base class in which it is declared. Thus, a protected member of a base
    class is accessible to any class within the class family, but not accessible to things
    outside the class family'; return false;">Protected member</a><bR>
<tr>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
    value='An informal set of English-like statements that are generally accepted within the
    computer industry to denote common computer programming operations. Pseudocode statements
    are used to describe the steps in a computer algorithm'; return false;">Pseudocode</a><bR
    >
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
    value='A base class that allows all of its public members to be public in its derived
    classes'; return false;">Public base class</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
    value='A primary memory storage structure that consists of a list, or sequence, of data
    elements. All insertions of elements ino the queue are made at one end of the queue,
    called the rear of the queue and all deletions of elements from the queue are made at the
    other end of the queue, called the front of the queue. A queue operates on the first-in,
    first-out, or FIFO principle'; return false;">Queue</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
    value='Reading is obtaining data from something such as an input device or a data
    structure, a copy operation. A read operation is usually a non-destructive operation';
    return false;">Reading</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
    value='A process whereby an operation calls itself until a primitive state is reached';
    return false;">Recursion</a><bR>
<tr>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
    value='A function that calls itself'; return false;">Recursive function</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
    value='Parameters that provide two-way communication between the clling program and the
    function'; return false;">Reference parameters</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
    value='When the program attempts to perform an illegal operation as defined by the laws
    of mathematics, logic, or the particular compiler in use'; return false;">Run-time error
    </a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
    value='Scope refers to the largest block in which a given constant or variable is
    accessible, or visible'; return false;">Scope</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
    value='A control structure where the program selects, or decides, between one of several
    routes depending on the conditions that are tested'; return false;">Selection</a><bR>
<tr>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
    value='A control structure where the statements are executed sequentially, one after
    another, in a straight-line fashion'; return false;">Sequence</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
    value='Single inheritance occurse when all inherited class members can be traced back to
    a single parent class'; return false;">Single inheritance</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
    value='The program that you write in the C++ language which normally has a file extension
    of .cpp'; return false;">Source program</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
    value='A primary memory storage structure that consists of a list, or sequence, of data
    elements where all the insertions and deletions of elements to and from the stack are
    made at one end of the stack called the top. A stack operates on the last-in, first-out,
    or LIFO principle'; return false;">Stack</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
    value='A predefined operation that the C++ compiler will recognize and evaluate to return
    a result or perform a given task'; return false;">Standard function</a><bR>
<tr>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
    value='In front of a pointer variable denotes the contents of the memory location where
    the pointer is pointing'; return false;">Star,*</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
    value='Static binding occurs when a polymorphic function is defined for several classes
    in a family and the actual code for the function is attached, or bound, at compile time.
    Overloaded functions are statically bound'; return false;">Static binding</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
    value='A collection of data members, or data fields, and function members. Also called a
    struct'; return false;">Structure</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.
    value='A methodology that requires software to be designed using a top/down modular
    approach'; return false;">Structured design</a><bR>
```

```
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='Structured programming allows programs to be written using well-defined control   ↙
    structures and independent program modules'; return false;">Structured programming</a><bR↙
    >
<tr>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='An error created by violating the required syntax, or grammar, of a programming   ↙
    language'; return false;">Syntax error</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='Parameters that allow for one-way communication of data from the calling program  ↙
    to the function'; return false;">Value parameters</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='Functions which have the same name but different implementations for various      ↙
    classes within a class family. A virtual function is dynamically bound'; return false;"> ↙
    Virtual function</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='Blanks, tabs, new lines, form feeds, etc. are all forms of whitespace'; return    ↙
    false;">Whitespace</a><bR>
<td><a href="definitions.html" onclick="javascript: parent.frames[1].document.box.definition.↙
    value='Writing is often associated with the output of data to a display monitor, printer ↙
    , or file. In addition, data can be written to data structures such as arrays, structs,  ↙
    or objects. A write operation is usually a destructive operation'; return false;">Writing↙
    </a><bR>
<tr>
</table>
</p>

<hr>
<br>
<p align="center">
<a href="http://www.bsu.edu/home.html" target="_top">
<img src="gifs/home.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="mailto:ASKBSU@bsu.edu" target="_top">
<img src="gifs/ask.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="http://www.bsu.edu/cgi-bin/guest" target="_top">
<img src="gifs/guest.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="http://www.bsu.edu/UP/about/bsu.html" target="_top">
<img src="gifs/about.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="http://www.bsu.edu/UP/map/indiana.html" target="_top">
<img src="gifs/map.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="http://www.bsu.edu/htmls/search.html" target="_top">
<img src="gifs/search.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="http://www.bsu.edu/home_help.html" target="_top">
<img src="gifs/help.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<br>
<font size="-1">
<a href="mailto:rawilson@bsuvc.bsu.edu">Author</a> |
Last Update: <SCRIPT>document.write(document.lastModified);</SCRIPT> |
<a href="mailto:rawilson@bsuvc.bsu.edu">Technical comments to the Webmaster
</a>
</font>
<br>
<font size="-4">Ball State University practices equal opportunity in education
and employment and is strongly and actively committed to diversity within its
community.
</font>
</p>

<p align="center">
<em>
<font size="-4">Links contained in this file to information provided
by organizations other than Ball State University's Computer Science
Department are presented as a service and neither constitute nor imply
endorsement or warranty.
</font>
</em>
</p>
```

```
</body>
</html>
```

```
<HTML>
<HEAD>
<!-- Author: Ryan Wilson -->
<SCRIPT>
// This function displays the time in the status line.
// Invoke it once to activate the clock; it will call itself from then on.
function display_time_in_status_line()
{
    var d = new Date();  // Get current time
    var h = d.getHours(); // Extract hours: 0 to 23
    var m = d.getMinutes(); // Extract minutes: 0 to 59
    var ampm = ( h >= 12)?"PM":"AM"; //Is it am or pm?
    if (h > 12)
        h-=12; //Convert 24-hour format to 12-hour
    if ( h == 0)
        h = 12; // Convert 0 o'clock to midnight
    if (m < 10)
        m = "0" + m;  //Convert 0 minutes to 00 minutes, etc.
    var t = h + ':' + m + ' ' + ampm;  // Put it all together

    defaultStatus = t;   //Display it in the status line.

    //Arrange to do it all again in 1 minute.
    setTimeout("display_time_in_status_line()", 60000); //60000 ms is 1 min.
}

var bcolor;
var allcookies = document.cookie;
var pos = allcookies.indexOf("color=");
if (pos != -1)
{
    var start = pos + 6;
    var end = allcookies.indexOf(";", start);
    if (end == -1)
        end = allcookies.length;
    var bcolor = allcookies.substring(start, end);
}
else
    bcolor = "white";

function setbgcolor(bcolor)
{
    document.bgColor = bcolor;
}
</SCRIPT>

</head>

<body text="#0D0F58" link="#760B10" alink="#544C51"
onLoad="display_time_in_status_line(); setbgcolor(bcolor);">

<FORM NAME="box">
<TEXTAREA NAME="definition" value="" ROWS=6 Cols=70 wrap=on></TEXTAREA>
</FORM>
</BODY>
</HTML>
```

C++ Operators

In order of precedence

Op	Name	Prec.	Assoc.	Arity	Overloadable	Example
::	Global scope resolution	17	Right	Unary	No	::a
::	Class scope resolution	17	Left	Binary	No	A::a
.	Direct member selection	16	Left	Binary	No	a.len
->	Indirect member selection	16	Left	Binary	Yes	a->len
[]	Subscript	16	Left	Binary	Yes	b[i]
()	Function call	16	Left	n/a	Yes	rand()
()	Type construction	16	Left	n/a	Yes	int(ch)
++	Post-increment	16	Right	Unary	Yes	c++
--	Post-decrement	16	Right	Unary	Yes	c--
sizeof	Size of object or type	15	Right	Unary	No	sizeof(a)
++	Pre-increment	15	Right	Unary	Yes	++a
--	Pre-decrement	15	Right	Unary	Yes	--a
~	Bitwise complement	15	Right	Unary	Yes	~v
!	Logical NOT	15	Right	Unary	Yes	!b
+	Unary plus	15	Right	Unary	Yes	+t
-	Unary Minus	15	Right	Unary	Yes	-t
*	Dereference	15	Right	Unary	Yes	*r
&	Address	15	Right	Unary	Yes	&g
new	Allocation	15	Right	Unary	Yes	new s
delete	Deallocation	15	Right	Unary	Yes	delete s
()	Type conversion	15	Right	Binary	Yes	int(ch)
.*	Direct member selection	14	Left	Binary	No	x.*s
->*	Indirect member selection	14	Left	Binary	Yes	p->*s
*	Multiplication	13	Left	Binary	Yes	a*b
/	Division	13	Left	Binary	Yes	a/b
%	Remainder	13	Left	Binary	Yes	a%b
+	Addition	12	Left	Binary	Yes	a + b
-	Subtraction	12	Left	Binary	Yes	a - b
<<	Bit shift left	11	Left	Binary	Yes	cout << a
>>	Bit shift right	11	Left	Binary	Yes	cin >> a

<	Less than	10	Left	Binary	Yes	a < b
<=	Les than or equal to	10	Left	Binary	Yes	a <= b
>	Greater than	10	Left	Binary	Yes	a > b
>=	Greater than or equal to	10	Left	Binary	Yes	a >= b
==	Equal to	9	Left	Binary	Yes	a == b
!=	Not equal to	9	Left	Binary	Yes	a != b
&	Bitwise AND	8	Left	Binary	Yes	a&b
^	Bitwise XOR	7	Left	Binary	Yes	a^b
|	Bitwise OR	6	Left	Binary	Yes	a|b
&&	Logical AND	5	Left	Binary	Yes	a && b
||	Logical OR	4	Left	Binary	Yes	a || b
?:	Conditional expression	3	Left	Ternary	No	a ? b : c
=	Assignment	2	Right	Binary	Yes	a = 17
+=	Addition assignment	2	Right	Binary	Yes	a += 2
-=	Subtraction assignment	2	Right	Binary	Yes	a -= 3
*=	Multiplication assignment	2	Right	Binary	Yes	a *= 4
/=	Division assignment	2	Right	Binary	Yes	a /= 20
%=	Remainder assignment	2	Right	Binary	Yes	a %= 3
&=	Bitwise AND assignment	2	Right	Binary	Yes	a &= mask
^=	Bitwise XOR assignment	2	Right	Binary	Yes	a ^= mask
|=	Bitwise OR assignment	2	Right	Binary	Yes	a |= mask
<<=	Bit shift left assignment	2	Right	Binary	Yes	a <<= 2
>>=	Bit shift right assignment	2	Right	Binary	Yes	a >>= 2
throw	Throw exception	1	Right	Unary	Yes	throw(14)
,	Comma	0	Left	Binary	Yes	--a,++b

```
<html>
<head>
<!-- Author: Ryan Wilson -->
<script>
// This function displays the time in the status line.
// Invoke it once to activate the clock; it will call itself from then on.
function display_time_in_status_line()
{
    var d = new Date();  // Get current time
    var h = d.getHours(); // Extract hours: 0 to 23
    var m = d.getMinutes(); // Extract minutes: 0 to 59
    var ampm = ( h >= 12)?"PM":"AM"; //Is it am or pm?
    if (h > 12)
        h-=12; //Convert 24-hour format to 12-hour
    if ( h == 0)
        h = 12; // Convert 0 o'clock to midnight
    if (m < 10)
        m = "0" + m;  //Convert 0 minutes to 00 minutes, etc.
    var t = h + ':' + m + ' ' + ampm;  // Put it all together

    defaultStatus = t;   //Display it in the status line.

    //Arrange to do it all again in 1 minute.
    setTimeout("display_time_in_status_line()", 60000); //60000 ms is 1 min.
}

// A variable we use to ensure that each error window we create is unique.
var error_count = 0;

// Email address to send the report to.
var email = "rawilson@bsuvc.bsu.edu";

// Define the error handler.  It generates an HTML form so the user
// can report the error to the author.
function report_error(msg, url, line)
{
    var w = window.open("","error"+error_count++,"resizable,status,width=625,height=400");
    // arguements... url (none specified), Name (force it to be unique), Features
    var d = w.document; // We use this variable to save typing.

    // Output an HTML document, including a form, into the new window.
    d.write('<DIV align=center>');
    d.write('<FONT SIZE=7 FACE="helvetica"><B>');
    d.write('OOPS.... A JavaScript Error Has Occurred!');
    d.write('</B></FONT><BR><HR SIZE=4 WIDTH="80%">');
    d.write('<FORM ACTION="mailto:' + email + '" METHOD=post');
    d.write(' ENCTYPE="text/plain">');
    d.write('<FONT SIZE=3>');
    d.write('<I>CLick the "Report Error" button to send a bug report.<I><BR>');
    d.write('<INPUT TYPE="submit" VALUE="Report Error">  ');
    d.write('<INPUT TYPE="button" VALUE="Dismiss" onClick="self.close()">');
    d.write('</DIV><DIV align=right>');
    d.write('<BR>Your name <I>(optional)</I>: ');
    d.write('<INPUT SIZE=42 NAME="name" VALUE="">');
    d.write('<BR>Error Message: ');
    d.write('<INPUT SIZE=42 NAME="message" VALUE="' + msg + '">');
    d.write('<BR>Document: <INPUT SIZE=42 NAME="url" VALUE="' + url + '">');
    d.write('<BR>Line Number: <INPUT SIZE=42 NAME="line" VALUE="' + line +'">');
    d.write('<BR>Browser Version: ');
    d.write('<INPUT SIZE=42 NAME="version" VALUE="'+navigator.userAgent + '">');
    d.write('</DIV></FONT>');
    d.write('</FORM>');
    // Remember to close the document when we're done.
    d.close();

    // Return true from this error handler, so that JavaScript does not
    // display it own error dialog.
    return true;
}

// Before the event handler can take effect, we have to register it
// for a particular window.
self.onerror = report_error;
```

```
var bcolor;
var allcookies = document.cookie;
var pos = allcookies.indexOf("color=");
if (pos != -1)
{
    var start = pos + 6;
    var end = allcookies.indexOf(";", start);
    if (end == -1)
        end = allcookies.length;
    var bcolor = allcookies.substring(start, end);
}
else
    bcolor = "white";

function setbgcolor(bcolor)
{
    document.bgColor = bcolor;
}
</SCRIPT>

<title>C++ Operators</title>
</head>

<body text="#0D0F58" link="#760B10" alink="#544C51"
onLoad="display_time_in_status_line(); setbgcolor(bcolor);">

<h2><center>C++ Operators</center></h2><BR>
<h3><center>In order of precedence</centeR></h3><BR>

<table border>
<th align=center>Op</th>
<th align=center>Name</th>
<th align=center>Prec.</th>
<th align=center>Assoc.</th>
<th align=center>Arity</th>
<th align=center>Overloadable</th>
<th align=center>Example</th>
<tr>
<Td>::</tD>
<tD>Global scope resolution</td>
<td>17</td>
<Td>Right</tD>
<tD>Unary</td>
<td>No</tD>
<Td>::a</tD>
<Tr>
<Td>::</td>
<Td>Class scope resolution</tD>
<Td>17</tD>
<Td>Left</tD>
<TD>Binary</tD>
<Td>No</tD>
<TD>A::a</td>
<Tr>
<Td>.</tD>
<td>Direct member selection</td>
<Td>16</tD>
<Td>Left</tD>
<Td>Binary</tD>
<Td>No</td>
<Td>a.len</tD>
<Tr>
<tD>-></tD>
<tD>Indirect member selection</tD>
<Td>16</tD>
<Td>Left</tD>
<Td>Binary</td>
<Td>Yes</tD>
<Td>a->len</tD>
<TR>
<Td>[]</tD>
<Td>Subscript</tD>
<TD>16</tD>
<Td>Left</td>
```

```
<Td>Binary</tD>
<Td>Yes</td>
<td>b[i]</tD>
<tr>
<TD>()</td>
<td>Function call</tD>
<Td>16</tD>
<td>Left</tD>
<TD>n/a</td>
<Td>Yes</tD>
<Td>rand()</tD>
<tr>
<td>()</tD>
<Td>Type construction</td>
<TD>16</td>
<Td>Left</tD>
<Td>n/a</td>
<td>Yes</tD>
<Td>int(ch)</tD>
<TR>
<td>++</tD>
<td>Post-increment</td>
<td>16</tD>
<td>Right</td>
<Td>Unary</tD>
<Td>Yes</td>
<Td>c++</tD>
<tr>
<td>--</tD>
<Td>Post-decrement</td>
<Td>16</td>
<Td>Right</tD>
<Td>Unary</tD>
<Td>Yes</tD>
<Td>c--</td>
<tr>
<Td>sizeof</tD>
<td>Size of object or type</tD>
<td>15</td>
<tD>Right</tD>
<Td>Unary</tD>
<td>No</tD>
<td>sizeof(a)</td>
<tr>
<Td>++</tD>
<Td>Pre-increment</tD>
<td>15</tD>
<td>Right</tD>
<Td>Unary</tD>
<TD>Yes</tD>
<Td>++a</tD>
<Tr>
<tD>--</tD>
<Td>Pre-decrement</tD>
<tD>15</tD>
<td>Right</td>
<td>Unary</tD>
<td>Yes</td>
<Td>--a</tD>
<tr>
<tD>~</tD>
<tD>Bitwise complement</tD>
<tD>15</td>
<td>Right</tD>
<Td>Unary</tD>
<td>Yes</td>
<tD>~v</td>
<Tr>
<Td>!</tD>
<Td>Logical NOT</tD>
<Td>15</tD>
<td>Right</tD>
<Td>Unary</tD>
<td>Yes</tD>
```

```
<Td>!b</td>
<Tr>
<td>+</tD>
<Td>Unary plus</tD>
<TD>15</tD>
<TD>Right</tD>
<Td>Unary</td>
<tD>Yes</tD>
<td>+t</tD>
<Tr>
<Td>-</tD>
<TD>Unary Minus</tD>
<TD>15</tD>
<TD>Right</td>
<td>Unary</td>
<td>Yes</tD>
<tD>-t</tD>
<tr>
<td>*</tD>
<td>Dereference</td>
<td>15</tD>
<Td>Right</tD>
<tD>Unary</tD>
<td>Yes</tD>
<Td>*r</tD>
<tr>
<Td>&</td>
<td>Address</tD>
<td>15</tD>
<td>Right</tD>
<Td>Unary</td>
<td>Yes</tD>
<tD>&g</tD>
<Tr>
<Td>new</tD>
<Td>Allocation</tD>
<TD>15</td>
<Td>Right</tD>
<Td>Unary</tD>
<Td>Yes</tD>
<td>new s</tD>
<tR>
<td>delete</tD>
<TD>Deallocation</tD>
<tD>15</tD>
<tD>Right</tD>
<tD>Unary</tD>
<Td>Yes</tD>
<Td>delete s</tD>
<Tr>
<TD>()</tD>
<Td>Type conversion</tD>
<TD>15</tD>
<Td>Right</tD>
<TD>Binary</tD>
<Td>Yes</tD>
<Td>int(ch)</tD>
<Tr>
<Td>.*</tD>
<Td>Direct member selection</td>
<TD>14</tD>
<Td>Left</tD>
<Td>Binary</tD>
<Td>No</tD>
<Td>x.*s</tD>
<tr>
<tD>->*</tD>
<TD>Indirect member selection</tD>
<Td>14</tD>
<tD>Left</tD>
<Td>Binary</tD>
<Td>Yes</tD>
<TD>p->*s</tD>
<TR>
```

```
<td>*</tD>
<tD>Multiplication</td>
<td>13</tD>
<td>Left</tD>
<Td>Binary</tD>
<tD>Yes</tD>
<TD>a*b</tD>
<tr>
<Td>/</tD>
<Td>Division</tD>
<td>13</tD>
<td>Left</tD>
<TD>Binary</tD>
<Td>Yes</tD>
<Td>a/b</tD>
<Tr>
<td>%</td>
<Td>Remainder</tD>
<Td>13</tD>
<Td>Left</tD>
<Td>Binary</tD>
<Td>Yes</tD>
<TD>a%b</tD>
<Tr>
<TD>+</tD>
<TD>Addition</tD>
<Td>12</tD>
<TD>Left</td>
<td>Binary</tD>
<td>Yes</tD>
<Td>a + b</tD>
<Tr>
<Td>-</tD>
<TD>Subtraction</tD>
<TD>12</tD>
<Td>Left</tD>
<TD>Binary</tD>
<tD>Yes</tD>
<td>a - b</tD>
<Tr>
<Td><<<</td>
<td>Bit shift left</tD>
<Td>11</tD>
<Td>Left</tD>
<TD>Binary</tD>
<Td>Yes</tD>
<Td>cout << a</tD>
<Tr>
<tD>>></tD>
<Td>Bit shift right</tD>
<TD>11</tD>
<Td>Left</tD>
<Td>Binary</tD>
<TD>Yes</td>
<td>cin >> a</tD>
<Tr>
<td><</tD>
<Td>Less than</td>
<tD>10</tD>
<Td>Left</tD>
<tD>Binary</tD>
<tD>Yes</tD>
<Td>a < b</tD>
<tr>
<td><=</tD>
<Td>Les than or equal to</tD>
<Td>10</tD>
<Td>Left</tD>
<Td>Binary</tD>
<td>Yes</tD>
<td>a <= b</tD>
<tr>
<td>></tD>
<Td>Greater than</tD>
```

```
<td>10</tD>
<td>Left</tD>
<Td>Binary</tD>
<TD>Yes</tD>
<Td>a > b</tD>
<tr>
<td>>=</tD>
<Td>Greater than or equal to</tD>
<Td>10</tD>
<TD>Left</tD>
<Td>Binary</tD>
<TD>Yes</tD>
<tD>a >= b</tD>
<tr>
<tD>==</tD>
<TD>Equal to</td>
<TD>9</tD>
<td>Left</td>
<TD>Binary</tD>
<TD>Yes</tD>
<tD>a == b</tD>
<TR>
<TD>!=</tD>
<Td>Not equal to</tD>
<TD>9</tD>
<td>Left</tD>
<tD>Binary</tD>
<tD>Yes</tD>
<TD>a != b</tD>
<Tr>
<tD>&</tD>
<TD>Bitwise AND</tD>
<td>8</tD>
<tD>Left</tD>
<tD>Binary</tD>
<TD>Yes</tD>
<tD>a&b</tD>
<Tr>
<Td>^</tD>
<TD>Bitwise XOR</tD>
<Td>7</tD>
<Td>Left</tD>
<TD>Binary</tD>
<tD>Yes</tD>
<TD>a^b</tD>
<tr>
<tD>|</tD>
<TD>Bitwise OR</tD>
<TD>6</tD>
<Td>Left</tD>
<tD>Binary</tD>
<TD>Yes</td>
<tD>a|b</tD>
<Tr>
<tD>&&</tD>
<Td>Logical AND</td>
<tD>5</tD>
<tD>Left</tD>
<tD>Binary</tD>
<TD>Yes</tD>
<Td>a && b</tD>
<tr>
<td>||</tD>
<TD>Logical OR</tD>
<Td>4</tD>
<Td>Left</tD>
<TD>Binary</tD>
<td>Yes</tD>
<tD>a || b</td>
<tr>
<td>?:</tD>
<TD>Conditional expression</tD>
<TD>3</tD>
<TD>Left</tD>
```

```
<TD>Ternary</tD>
<TD>No</tD>
<TD>a ? b : c</tD>
<Tr>
<tD>=</tD>
<Td>Assignment</tD>
<tD>2</tD>
<Td>Right</tD>
<td>Binary</tD>
<Td>Yes</tD>
<TD>a = 17</td>
<tR>
<Td>+=</tD>
<tD>Addition assignment</td>
<tD>2</tD>
<TD>Right</tD>
<TD>Binary</tD>
<TD>Yes</tD>
<Td>a += 2</tD>
<tr>
<tD>-=</tD>
<tD>Subtraction assignment</tD>
<TD>2</tD>
<Td>Right</tD>
<TD>Binary</tD>
<TD>Yes</tD>
<Td>a -= 3</tD>
<Tr>
<td>*=</tD>
<TD>Multiplication assignment</tD>
<tD>2</tD>
<Td>Right</td>
<td>Binary</tD>
<TD>Yes</tD>
<TD>a *= 4</td>
<tr>
<tD>/=</tD>
<tD>Division assignment</tD>
<tD>2</tD>
<TD>Right</td>
<tD>Binary</tD>
<tD>Yes</tD>
<td>a /= 20</tD>
<tr>
<tD>%=</tD>
<td>Remainder assignment</tD>
<Td>2</tD>
<td>Right</tD>
<tD>Binary</tD>
<td>Yes</tD>
<TD>a %= 3</tD>
<Tr>
<TD>&=</tD>
<TD>Bitwise AND assignment</tD>
<TD>2</tD>
<td>Right</tD>
<tD>Binary</tD>
<td>Yes</tD>
<td>a &= mask</tD>
<Tr>
<td>^=</tD>
<tD>Bitwise XOR assignment</tD>
<TD>2</tD>
<Td>Right</tD>
<TD>Binary</tD>
<td>Yes</tD>
<TD>a ^= mask</tD>
<tR>
<tD>|=</tD>
<td>Bitwise OR assignment</tD>
<tD>2</tD>
<td>Right</tD>
<tD>Binary</tD>
<tD>Yes</tD>
```

```
<TD>a |= mask</tD>
<tr>
<tD><<=</tD>
<tD>Bit shift left assignment</tD>
<td>2</tD>
<tD>Right</td>
<tD>Binary</td>
<tD>Yes</tD>
<td>a <<= 2</tD>
<tr>
<tD>>>=</tD>
<tD>Bit shift right assignment</tD>
<tD>2</tD>
<tD>Right</tD>
<TD>Binary</tD>
<TD>Yes</tD>
<TD>a >>= 2</tD>
<TR>
<Td>throw</tD>
<td>Throw exception</tD>
<TD>1</tD>
<TD>Right</tD>
<td>Unary</tD>
<td>Yes</tD>
<tD>throw(14)</tD>
<Tr>
<tD>,</tD>
<td>Comma</tD>
<TD>0</tD>
<tD>Left</Td>
<tD>Binary</td>
<tD>Yes</tD>
<tD>--a,++b</tD>
<tR>
</table>

<hr>
<br>
<p align="center">
<a href="http://www.bsu.edu/home.html" target="_top">
<img src="gifs/home.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="mailto:ASKBSU@bsu.edu" target="_top">
<img src="gifs/ask.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="http://www.bsu.edu/cgi-bin/guest" target="_top">
<img src="gifs/guest.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="http://www.bsu.edu/UP/about/bsu.html" target="_top">
<img src="gifs/about.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="http://www.bsu.edu/UP/map/indiana.html" target="_top">
<img src="gifs/map.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="http://www.bsu.edu/htmls/search.html" target="_top">
<img src="gifs/search.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="http://www.bsu.edu/home_help.html" target="_top">
<img src="gifs/help.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<br>
<font size="-1">
<a href="mailto:rawilson@bsuvc.bsu.edu">Author</a> |
Last Update: <SCRIPT>document.write(document.lastModified);</SCRIPT> |
<a href="mailto:rawilson@bsuvc.bsu.edu">Technical comments to the Webmaster</a>
</font>
<br>
<font size="-4">Ball State University practices equal opportunity in education
and employment and is strongly and actively committed to diversity within its
community.
</font>
</p>

<p align="center">
```

```
<em>
<font size="-4">Links contained in this file to information provided
by organizations other than Ball State University's Computer Science Department
are presented as a service and neither constitute nor imply endorsement or
warranty.
</font>
</em>
</p>
</body>
</html>
```

Other C++ Resources on the Web

BSU CS Dept.
BSU ACM chapter
Code Guru
A Coding Convention for C++ Code
Microsoft
The Bits Editors
navigator for C++
The C++ Programming Language
Jon's C++ Resources Directory
c++ resources
Ask the C++ Pro
Tips for Visual C++
Association of C and C++ Users
tutorials and resources
Cprogramming.com
The Development of the C Language
Programing pages
From The Ground Up: A Guide to C++
Quadralay's C++ Archive
C/C++
C++ Report
C and C++ Links
C++ FAQ LITE
Programming in C
The Ground Cero Guide To C
programming.c
CodeHelp.com
BOFH

```
<html>
<head>
<!-- Author: Ryan Wilson -->
<script>
// This function displays the time in the status line.
// Invoke it once to activate the clock; it will call itself from then on.
function display_time_in_status_line()
{
    var d = new Date();  // Get current time
    var h = d.getHours(); // Extract hours: 0 to 23
    var m = d.getMinutes(); // Extract minutes: 0 to 59
    var ampm = ( h >= 12)?"PM":"AM"; //Is it am or pm?
    if (h > 12)
        h-=12; //Convert 24-hour format to 12-hour
    if ( h == 0)
        h = 12; // Convert 0 o'clock to midnight
    if (m < 10)
        m = "0" + m;  //Convert 0 minutes to 00 minutes, etc.
    var t = h + ':' + m + ' ' + ampm;  // Put it all together

    defaultStatus = t;   //Display it in the status line.

    //Arrange to do it all again in 1 minute.
    setTimeout("display_time_in_status_line()", 60000); //60000 ms is 1 min.
}

// A variable we use to ensure that each error window we create is unique.
var error_count = 0;

// Email address to send the report to.
var email = "rawilson@bsuvc.bsu.edu";

// Define the error handler.  It generates an HTML form so the user
// can report the error to the author.
function report_error(msg, url, line)
{
    var w = window.open("","error"+error_count++,"resizable,status,width=625,height=400");
    // arguements... url (none specified), Name (force it to be unique), Features
    var d = w.document; // We use this variable to save typing.

    // Output an HTML document, including a form, into the new window.
    d.write('<DIV align=center>');
    d.write('<FONT SIZE=7 FACE="helvetica"><B>');
    d.write('OOPS.... A JavaScript Error Has Occurred!');
    d.write('</B></FONT><BR><HR SIZE=4 WIDTH="80%">');
    d.write('<FORM ACTION="mailto:' + email + '" METHOD=post');
    d.write(' ENCTYPE="text/plain">');
    d.write('<FONT SIZE=3>');
    d.write('<I>CLick the "Report Error" button to send a bug report.<I><BR>');
    d.write('<INPUT TYPE="submit" VALUE="Report Error">  ');
    d.write('<INPUT TYPE="button" VALUE="Dismiss" onClick="self.close()">');
    d.write('</DIV><DIV align=right>');
    d.write('<BR>Your name <I>(optional)</I>: ');
    d.write('<INPUT SIZE=42 NAME="name" VALUE="">');
    d.write('<BR>Error Message: ');
    d.write('<INPUT SIZE=42 NAME="message" VALUE="' + msg + '">');
    d.write('<BR>Document: <INPUT SIZE=42 NAME="url" VALUE="' + url + '">');
    d.write('<BR>Line Number: <INPUT SIZE=42 NAME="line" VALUE="' + line +'">');
    d.write('<BR>Browser Version: ');
    d.write('<INPUT SIZE=42 NAME="version" VALUE="'+navigator.userAgent + '">');
    d.write('</DIV></FONT>');
    d.write('</FORM>');
    // Remember to close the document when we're done.
    d.close();

    // Return true from this error handler, so that JavaScript does not
    // display it own error dialog.
    return true;
}

// Before the event handler can take effect, we have to register it
// for a particular window.
self.onerror = report_error;
```

```
var bcolor;
var allcookies = document.cookie;
var pos = allcookies.indexOf("color=");
if (pos != -1)
{
    var start = pos + 6;
    var end = allcookies.indexOf(";", start);
    if (end == -1)
        end = allcookies.length;
    var bcolor = allcookies.substring(start, end);
}
else
    bcolor = "white";

function setbgcolor(bcolor)
{
    document.bgColor = bcolor;
}
</SCRIPT>

<title>Other C++ Resources on the Web</title>
</head>

<body text="#0D0F58" link="#760B10" alink="#544C51"
onLoad="display_time_in_status_line(); setbgcolor(bcolor);">

<h2><center>Other C++ Resources on the Web</center></h2><br>

<center>
<a href="http://www.cs.bsu.edu/" target="_blank">BSU CS Dept.</a><br>
<a href="http://www.cs.bsu.edu/~acm/" target="_blank">BSU ACM chapter</a><Br>
<a href="http://www.codeguru.com/" target="_blank">Code Guru</a><bR>
<a href="http://www.cs.rice.edu/~dwallach/CPlusPlusStyle.html" target="_blank">A Coding ↙
    Convention for C++ Code</a><br>
<a href="http://msdn.microsoft.com/VISUALC/" target="_blank">Microsoft</a><br>
<a href="http://www.richplum.co.uk/cbuilder/" target="_blank">The Bits Editors</a><br>
<a href="http://www.topcode.com/c++/main.shtml" target="_blank">navigator for C++</a><br>
<a href="http://www.research.att.com/~bs/C++.html" target="_blank">The C++ Programming ↙
    Language</a><br>
<a href="http://www.cs.bham.ac.uk/~jdm/cpp.html" target="_blank">Jon's C++ Resources ↙
    Directory</a><br>
<a href="http://www.cera2.com/softd/cplus.htm" target="_blank">c++ resources</a><br>
<a href="http://www.inquiry.com/techtips/cpp_pro/" target="_blank">Ask the C++ Pro</a><br>
<a href="http://www.pinpub.com/vcd/tips.htm" target="_blank">Tips for Visual C++</a><br>
<a href="http://www.accu.org/" target="_blank">Association of C and C++ Users</a><br>
<a href="http://www.gustavo.net/programming/c.shtml" target="_blank">tutorials and resources ↙
    </a><br>
<a href="http://www.cprogramming.com/" target="_blank">Cprogramming.com</a><br>
<a href="http://cm.bell-labs.com/cm/cs/who/dmr/chist.html" target="_blank">The Development of↙
    the C Language</a><br>
<a href="http://www.netlink.co.uk/users/zion/comp/prog.htm" target="_blank">Programing pages ↙
    </a><br>
<a href="http://library.advanced.org/3074/" target="_blank">From The Ground Up: A Guide to C ↙
    ++</a><br>
<a href="http://www.austinlinks.com/CPlusPlus/" target="_blank">Quadralay's C++ Archive</a>< ↙
    br>
<a href="http://cplus.miningco.com/index.htm" target="_blank">C/C++</a><br>
<a href="http://www.creport.com/" target="_blank">C++ Report</a><br>
<a href="http://walden.mo.net/~mikemac/clink.html" target="_blank">C and C++ Links</a><br>
<a href="http://www.cerfnet.com/~mpcline/C++-FAQs-Lite/" target="_blank">C++ FAQ LITE</a><br>
<a href="http://www.cm.cf.ac.uk/Dave/C/CE.html" target="_blank">Programming in C</a><br>
<a href="http://home1.inet.tele.dk/seth/c-tut.html" target="_blank">The Ground Cero Guide To ↙
    C</a><br>
<a href="http://www.flamingolingo.com/programming.c/" target="_blank">programming.c</a><br>
<a href="http://codehelp.com/" target="_blank">CodeHelp.com</a><br>
<a href="http://www.iinet.net.au/~bofh/" target="_blank">BOFH</a><br>
<hr>
<br>
<p align="center">
<a href="http://www.bsu.edu/home.html" target="_top">
<img src="gifs/home.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="mailto:ASKBSU@bsu.edu" target="_top">
<img src="gifs/ask.gif" border="0" WIDTH="72" HEIGHT="14">
```

```
</a>
<a href="http://www.bsu.edu/cgi-bin/guest" target="_top">
<img src="gifs/guest.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="http://www.bsu.edu/UP/about/bsu.html" target="_top">
<img src="gifs/about.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="http://www.bsu.edu/UP/map/indiana.html" target="_top">
<img src="gifs/map.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="http://www.bsu.edu/htmls/search.html" target="_top">
<img src="gifs/search.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="http://www.bsu.edu/home_help.html" target="_top">
<img src="gifs/help.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<br>
<font size="-1">
<a href="mailto:rawilson@bsuvc.bsu.edu">Author</a> |
Last Update: <SCRIPT>document.write(document.lastModified);</SCRIPT> |
<a href="mailto:rawilson@bsuvc.bsu.edu">Technical comments to the Webmaster</a>
</font>
<br>
<font size="-4">Ball State University practices equal opportunity in education
and employment and is strongly and actively committed to diversity within its
community.
</font>
</p>

<p align="center">
<em>
<font size="-4">Links contained in this file to information provided
by organizations other than Ball State University's Computer Science Department
are presented as a service and neither constitute nor imply endorsement or
warranty.
</font>
</em>
</p>
</body>
</html>
```

Please use the form to submit suggestions and/or bug reports to the author of the CS 120 Resource site. All information requested on this form is optional but it would be helpful to receive answers to the two questions. Also, if you would like a response to your submission then don't forget to include your name and email address. Thanks for your help!

First Name:
Last Name:
Email:
Are you currently enrolled in CS120?
○ Yes ○ No
Did you find this site useful?
○ Yes ○ No
Please use the box below for comments or error reports.

Reset Submit

```
<html>
<head>
<!-- Author: Ryan Wilson -->
<script>
// This function displays the time in the status line.
// Invoke it once to activate the clock; it will call itself from then on.
function display_time_in_status_line()
{
    var d = new Date();  // Get current time
    var h = d.getHours(); // Extract hours: 0 to 23
    var m = d.getMinutes(); // Extract minutes: 0 to 59
    var ampm = ( h >= 12)?"PM":"AM"; //Is it am or pm?
    if (h > 12)
        h-=12; //Convert 24-hour format to 12-hour
    if ( h == 0)
        h = 12; // Convert 0 o'clock to midnight
    if (m < 10)
        m = "0" + m;  //Convert 0 minutes to 00 minutes, etc.
    var t = h + ':' + m + ' ' + ampm;  // Put it all together
    defaultStatus = t;   //Display it in the status line.
    //Arrange to do it all again in 1 minute.
    setTimeout("display_time_in_status_line()", 60000); //60000 ms is 1 min.
}

// A variable we use to ensure that each error window we create is unique.
var error_count = 0;
// Email address to send the report to.
var email = "rawilson@bsuvc.bsu.edu";
// Define the error handler.  It generates an HTML form so the user
// can report the error to the author.
function report_error(msg, url, line)
{
    var w = window.open("","error"+error_count++,"resizable,status,width=625,height=400");
    // arguments... url (none specified), Name (force it to be unique), Features
    var d = w.document; // We use this variable to save typing.
    // Output an HTML document, including a form, into the new window.
    d.write('<DIV align=center>');
    d.write('<FONT SIZE=7 FACE="helvetica"><B>');
    d.write('OOPS.... A JavaScript Error Has Occurred!');
    d.write('</B></FONT><BR><HR SIZE=4 WIDTH="80%">');
    d.write('<FORM ACTION="mailto:' + email + '" METHOD=post');
    d.write(' ENCTYPE="text/plain">');
    d.write('<FONT SIZE=3>');
    d.write('<I>CLick the "Report Error" button to send a bug report.<I><BR>');
    d.write('<INPUT TYPE="submit" VALUE="Report Error">  ');
    d.write('<INPUT TYPE="button" VALUE="Dismiss" onClick="self.close()">');
    d.write('</DIV><DIV align=right>');d.write('<BR>Your name <I>(optional)</I>: ');
    d.write('<INPUT SIZE=42 NAME="name" VALUE="">');
    d.write('<BR>Error Message: ');
    d.write('<INPUT SIZE=42 NAME="message" VALUE="' + msg + '">');
    d.write('<BR>Document: <INPUT SIZE=42 NAME="url" VALUE="' + url + '">');
    d.write('<BR>Line Number: <INPUT SIZE=42 NAME="line" VALUE="' + line +'">');
    d.write('<BR>Browser Version: ');
    d.write('<INPUT SIZE=42 NAME="version" VALUE="'+navigator.userAgent + '">');
    d.write('</DIV></FONT>');
    d.write('</FORM>');
    // Remember to close the document when we're done.
    d.close();
    // Return true from this error handler, so that JavaScript does not
    // display it own error dialog.
    return true;
}
// Before the event handler can take effect, we have to register it
// for a particular window.
self.onerror = report_error;
var bcolor;
var allcookies = document.cookie;
var pos = allcookies.indexOf("color=");
if (pos != -1)
{
    var start = pos + 6;
    var end = allcookies.indexOf(";", start);
    if (end == -1)
        end = allcookies.length;
```

```
    var bcolor = allcookies.substring(start, end);
}
else
    bcolor = "white";
function setbgcolor(bcolor)
{
    document.bgColor = bcolor;
}
</SCRIPT>
</head>
<body text="#0D0F58" link="#760B10" alink="#544C51"
onLoad="display_time_in_status_line(); setbgcolor(bcolor);">
<p>Please use the form to submit suggestions and/or bug reports to the author of
the CS 120 Resource site.  All information requested on this form is optional
but it would be helpful to receive answers to the two questions.  Also, if
you would like a response to your submission then don't forget to include
your name and email address.  Thanks for your help!
</p>
<FORM method=Post NAME="suggestion" ACTION="mailto:raw@bsu-cs.bsu.edu"
enctype="text/plain">
<!-- onsubmit="parent.frames[1].URL='suggestion2.html';parent.frames[1].location.reload();
return true;"-->
First Name: <INPUT TYPE=text NAME="fname" SIZE=15>
<br>
Last Name: <INPUT TYPE=text NAME="lname" SIZE=15>
<br>
Email: <INPUT TYPE=text NAME="email" SIZE=50><br>
Are you currently enrolled in CS120?
<br>
<INPUT TYPE=radio NAME="cs120" VALUE="yes">Yes
<INPUT TYPE=radio NAME="cs120" VALUE="no">No
<br>
Did you find this site useful?
<bR>
<INPUT TYPE=radio NAME="useful" VALUE="yes">Yes
<INPUT TYPE=radio NAME="useful" VALUE="no">No<br>
Please use the box below for comments or error reports.
<br>
<TEXTAREA NAME="comments" ROWS=10 COLS=70>
</TEXTAREA>
<br>
<center>
<input TYPE="reset" VALUE="Reset" name="reset1">
<input TYPE="submit" VALUE="Submit" name="submit1">
<br>
</center>
</FORM>

<!-- footer ->
<hr>
<br>
<p align="center">
<a href="http://www.bsu.edu/home.html" target="_top">
<img src="gifs/home.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="mailto:ASKBSU@bsu.edu" target="_top">
<img src="gifs/ask.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="http://www.bsu.edu/cgi-bin/guest" target="_top">
<img src="gifs/guest.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="http://www.bsu.edu/UP/about/bsu.html" target="_top">
<img src="gifs/about.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="http://www.bsu.edu/UP/map/indiana.html" target="_top">
<img src="gifs/map.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="http://www.bsu.edu/htmls/search.html" target="_top">
<img src="gifs/search.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="http://www.bsu.edu/home_help.html" target="_top">
<img src="gifs/help.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<br>
```

```
<font size="-1">
<a href="mailto:rawilson@bsuvc.bsu.edu">Author</a> |
Last Update: <SCRIPT>document.write(document.lastModified);</SCRIPT> |
<a href="mailto:rawilson@bsuvc.bsu.edu">Technical comments to the Webmaster</a>
</font>
<br>
<font size="-4">Ball State University practices equal opportunity in education
and employment and is strongly and actively committed to diversity within its
community.
</font>
</p>
<p align="center">
<em>
<font size="-4">Links contained in this file to information provided
by organizations other than Ball State University's Computer Science Department
are presented as a service and neither constitute nor imply endorsement or
warranty.
</font>
</em>
</p>
</body>
</html>
```

Common UNIX Commands

Since Unix and Linux use similar commands, these commands will also run on the machines in RB 369.
Note that the % symbol denotes the prompt.

`% passwd`	allows you to change your password; follow instructions on screen
`% ls`	displays the contents of current directory in short format
`% ls -la`	displays all files in current directory, including the hidden files, in long format with size, privileges, owner, dates, times, etc.
`% cat filename`	displays the contents of the file
`% more filename`	displays the contents of the file one screen at a time; to advance the page, enter a return; to stop viewing, enter a q for quit
`% cp file1 file2`	makes a copy of file1 and names it file2
`% mv file1 file2`	renames file1 to file2
`% rm filename`	deletes the file named -- be sure you want to do this!
`% lpr -Prb369 filename`	sends a copy of file1 to the printer in RB369
`% lpq -Prb369`	lists the contents of the printer queue for printer in RB 369 (printers are named for the room in which they are located)
`% mkdir dirname`	creates a subdirectory of current directory
`% cd`	changes current directory to your home directory
`% cd dirname`	changes current directory to the named directory
`% cd ..`	changes current directory to the next higher level directory
`% pwd`	prints the name of the current working directory
`% man` *cmdname*	gives explanation for the given Unix command
`% man -k` *topic*	gives help for finding correct Unix command(s) related to *topic*
`% netscape`	runs the netscape browser to the World Wide Web
`% elm`	runs the program which allows you to send and read mail -- do this only from *bsu-cs*
`% turnin filename`	submits your file to the appropriate account for grading. It is imperative that you do this only from *bsu-cs*
`% g++ -s -o pg.exe pg.cc`	compiles the file named pg.cc and either creates an executable file named pg.exe or gives a listing of the errors
`% file.exe`	executes the program file.exe
`% hostname`	displays the name of the machine of which you are logged into
`% telnet bsu-cs`	connects you to *bsu-cs*, should be used from workstation
`% rlogin bsu-cs -l` *username*	connects the user with specified name to *bsu-cs*
`% logout`	logs you off from *bsu-cs* (does not apply to workstations!) Control-D also disconnects you.
`% quota`	displays the amount of resources alloted to your account and the amount of resources used.

```
<HTML>
<HEAD>
<! Authors: Dr. Bonita McVey & Ryan Wilson -->
<META HTTP-EQUIV="Content-Type" CONTENT="text/html; charset=ISO-8859-1">
<META NAME="GENERATOR" CONTENT="Internet Assistant for Microsoft Word 2.04z">
<TITLE>Common UNIX Commands</TITLE>
</HEAD>
<BODY>
<P><CENTER><B><Font Size=4>Common UNIX Commands</B></CENTER></P>
<center>
Since Unix and Linux use similar commands, these commands will also run on
the machines in RB 369. <br> Note that the % symbol denotes the prompt.
</center>
<hr>
<TABLE>
<TR><TD WIDTH=235><TT>% passwd</TT></TD><TD WIDTH=355><FONT SIZE=2>allows you to change your
    password; follow instructions on screen</FONT>
</TD></TR>
<TR><TD WIDTH=235><TT>% ls</TT></TD><TD WIDTH=355><FONT SIZE=2>displays the contents of
    current directory in short format</FONT>
</TD></TR>
<TR><TD WIDTH=235><TT>% ls -la</TT></TD><TD WIDTH=355><FONT SIZE=2>displays all files in
    current directory, including the hidden files, in long format with size, privileges,
    owner, dates, times, etc.</FONT>
</TD></TR>
<TR><TD WIDTH=235><TT>% cat filename</TT></TD><TD WIDTH=355><FONT SIZE=2>displays the
    contents of the file</FONT>
</TD></TR>
<TR><TD WIDTH=235><TT>% more filename</TT></TD><TD WIDTH=355><FONT SIZE=2>displays the
    contents of the file one screen at a time; to advance the page, enter a return; to stop
    viewing, enter a q for quit </FONT>
</TD></TR>
<TR><TD WIDTH=235><TT>% cp file1 file2</TT></TD><TD WIDTH=355><FONT SIZE=2>makes a copy of
    file1 and names it file2</FONT>
</TD></TR>
<TR><TD WIDTH=235><TT>% mv file1 file2</TT></TD><TD WIDTH=355><FONT SIZE=2>renames file1 to
    file2</FONT>
</TD></TR>
<TR><TD WIDTH=235><TT>% rm filename</TT></TD><TD WIDTH=355><FONT SIZE=2>deletes the file
    named -- be sure you want to do this!</FONT>
</TD></TR>
<TR><TD WIDTH=235><TT>% lpr -Prb369</TT> <TT>filename</TT>  </FONT>
</TD><TD WIDTH=355><FONT SIZE=2>sends a copy of file1 to the printer in RB369</FONT>
</TD></TR>
<TR><TD WIDTH=235><TT>% lpq -Prb369</TT><FONT SIZE=2> </FONT>
</TD><TD WIDTH=355><FONT SIZE=2>lists the contents of the printer queue for printer in RB 369
    </FONT> <FONT SIZE=2>(printers are named for the room in which they are located)</FONT>
</TD></TR>
<TR><TD WIDTH=235><TT>% mkdir dirname</TT><FONT SIZE=2> </FONT>
</TD><TD WIDTH=355><FONT SIZE=2>creates a subdirectory of current directory</FONT>
</TD></TR>
<TR><TD WIDTH=235><TT>% cd</TT></TD><TD WIDTH=355><FONT SIZE=2>changes current directory to
    your home directory</FONT>
</TD></TR>
<TR><TD WIDTH=235><TT>% cd dirname</TT></TD><TD WIDTH=355><FONT SIZE=2>changes current
    directory to the named directory</FONT>
</TD></TR>
<TR><TD WIDTH=235><TT>% cd ..</TT><FONT SIZE=2> </FONT></TD><TD WIDTH=355><FONT SIZE=2>
    changes current directory to the next higher level directory</FONT>
</TD></TR>
<TR><TD WIDTH=235><TT>% pwd</TT></TD><TD WIDTH=355><FONT SIZE=2>prints the name of the
    current working directory</FONT>
</TD></TR>
<TR><TD WIDTH=235><TT>% man</TT><FONT SIZE=2> <I>cmdname</I> </FONT>
</TD><TD WIDTH=355><FONT SIZE=2>gives explanation for the given Unix command</FONT>
</TD></TR>
<TR><TD WIDTH=235><TT>% man -k</TT><FONT SIZE=2> <I>topic</I> </FONT>
</TD><TD WIDTH=355><FONT SIZE=2>gives help for finding correct Unix command(s) related to <I>
    topic</I></FONT>
</TD></TR>
<TR><TD WIDTH=235><TT>% netscape</TT><FONT SIZE=2> </FONT></TD>
<TD WIDTH=355><FONT SIZE=2>runs the netscape browser to the World Wide Web</FONT>
</TD></TR>
<TR><TD WIDTH=235><TT>% elm</TT><FONT SIZE=2> </FONT></TD><TD WIDTH=355><FONT SIZE=2>runs the
```

```
    program which allows you to send and read mail -- do this only from <I>bsu-cs</I></FONT>
</TD></TR>
<TR><TD WIDTH=235><TT>% turnin filename</TT></TD><TD WIDTH=355><FONT SIZE=2>submits your file
    to the appropriate account for grading.  It is imperative that you do this only from <I>
    bsu-cs</I></FONT>
</TD></TR>
<TR><TD WIDTH=235><TT>% g++ -s -o pg.exe pg.cc</TT></TD><TD WIDTH=355><FONT SIZE=2>compiles
    the file named pg.cc and either creates an executable file named pg.exe or gives a
    listing of the errors</FONT>
</TD></TR>
<TR><TD WIDTH=235><TT>% file.exe</TT><FONT SIZE=2> </FONT></TD>
<TD WIDTH=355><FONT SIZE=2>executes the program file.exe</FONT>
</TD></TR>
<TR><TD WIDTH=235><TT>% hostname</TT><FONT SIZE=2> </FONT></TD>
<TD WIDTH=355><FONT SIZE=2>displays the name of the machine of which you are logged into</
    FONT>
</TD></TR>
<TR><TD WIDTH=235><TT>% telnet bsu-cs</TT><FONT SIZE=2> </FONT>
</TD><TD WIDTH=355><FONT SIZE=2>connects you to <i>bsu-cs</i>, should be used from
    workstation</FONT>
</TD></TR>
<TR><TD WIDTH=235><TT>% rlogin bsu-cs -l <i>username</i></TT></TD><TD WIDTH=355><FONT SIZE=2>
    connects the user with specified name to <i>bsu-cs</i> </FONT>
</TD></TR>
<TR><TD WIDTH=235><TT>% logout</TT></TD><TD WIDTH=355><FONT SIZE=2>logs you off from <i>
    bsu-cs</i> (does not apply to workstations!)</FONT> <FONT SIZE=2>Control-D also
    disconnects you.</FONT>
</TD></TR>
<TR><TD WIDTH=235><TT>% quota </TT></TD><TD WIDTH=355><FONT SIZE=2>displays the amount of
    resources alloted to your account and the amount of resources used.
</TD></TR>
</TABLE>
<P>
<hr>
<br>
<p align="center">
<a href="http://www.bsu.edu/home.html" target="_top">
<img src="gifs/home.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="mailto:ASKBSU@bsu.edu" target="_top">
<img src="gifs/ask.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="http://www.bsu.edu/cgi-bin/guest" target="_top">
<img src="gifs/guest.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="http://www.bsu.edu/UP/about/bsu.html" target="_top">
<img src="gifs/about.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="http://www.bsu.edu/UP/map/indiana.html" target="_top">
<img src="gifs/map.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="http://www.bsu.edu/htmls/search.html" target="_top">
<img src="gifs/search.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<a href="http://www.bsu.edu/home_help.html" target="_top">
<img src="gifs/help.gif" border="0" WIDTH="72" HEIGHT="14">
</a>
<br>
<font size="-1">
<a href="mailto:rawilson@bsuvc.bsu.edu">Author</a> |
Last Update: <SCRIPT>document.write(document.lastModified);</SCRIPT> |
<a href="mailto:rawilson@bsuvc.bsu.edu">Technical comments to the Webmaster</a>
</font>
<br>
<font size="-4">
Ball State University practices equal opportunity in education and employment
and is strongly and actively committed to diversity within its community.
</font>
</p>

<p align="center">
<em>
<font size="-4">
Links contained in this file to information provided
```

```
by organizations other than Ball State University's Computer Science
Department are presented as a service and neither constitute nor imply
endorsement or warranty.
</font>
</em>
</p>
</BODY>
<HTML>
```

The Computer Science UNIX System

The Unix computer system, known as *bsu-cs*, is a SunUltra Server 170 computer connected to the university ethernet backbone, has the internet address of 147.226.112.101 and a full domain name of *bsu-cs.bsu.edu.*

How to Login to the *bsu-cs* System

Once you have an account and have connected to the *bsu-cs* system, enter your login name and password at the appropriate prompts. Your password will not appear on the screen as you enter it in order to keep your password secure from others. The screen will look similar to this:

```
Unix(r) System V Release 4.0 (bsu-cs)

login: your_login_name<return>
password: your_password<return>
```

Once you have entered your login name and password correctly, the system will display the login messages of the day. Please read these carefully. Any system changes, downtimes, notices, etc., will be shown here.

How to Connect to and Disconnect from *bsu-cs*

You will be able to connect to *bsu-cs* from various locations including RB 369, 355, 356, 134, the student modem pool, from your VAX account, etc. The methods used in each location varies and are described below. Be sure to gain experience with each of the CS lab rooms as it is possible that RB 369 will be crowded when open.

RB 369: When you sit down at a machine in RB 369, you should see a screen that says "**Welcome to *machine-name***" . Be sure to select (click on) `Session`, then select `Default`. Enter your username and password for your Unix account. Press the right mouse button on the desktop to activate a pop-up menu. From the menu, choose `Add`, then choose `Terminal`. This will open a terminal window to the local workstation (the one at which you are sitting). You may open additional windows by again using the right mouse button and the same sequence of steps. To connect to *bsu-cs*, you may enter either the command `rlogin bsu-cs` or the command `telnet bsu-cs` .

To disconnect from *bsu-cs*, you need only enter the command `logout`; however, you still need to disconnect from the local workstation. To do so, click on the **Gnome foot** in the lower left hand corner of the screen and choose **Logout**. From the window that opens, choose **Logout** and click **Yes**.

DO NOT TURN OFF THE MACHINES IN RB355, RB356 or RB369!

RB 355/356: When you sit down, you should be looking at a screen with a box that has a **User name**, in which you should enter `RB355` or `RB356`, depending on your location. After doing so, simply click on **OK** or type the return key. Shortly there will appear several icons and a button in lower left corner labeled **Start.** Click on this button and choose **Programs** from the menu that appears. Another menu will appear from which you choose **Telnet**. Finally there are two options with Telnet, *bsu-cs* and VAX. Choose *bsu-cs.* You are now connected to *bsu-cs*.

To disconnect from bsu-cs, first be sure that you have logged out. Then, go to the **Start** button, select

ShutDown. Another menu will appear from which you should choose to **Close all programs and log on as a different user**. Then click on **Yes.**
DO NOT TURN OFF THE MACHINES IN RB355, RB356 or RB369!

RB 134: Choose the option U for Unix from the menu that appears at the top of the screen and then type return. You may now login. Logging out from your account will disconnect the machine from *bsu-cs.*

Dial-up: Choose option 2, the Unix system and then type return.

Mail

Mail is one of the most commonly used applications on any multi-user computer. The Unix system currently has two mail applications, *mail* and *elm*. *Mail* is the standard issue for a Unix system and *elm* is a public domain software package that is menu driven and more screen oriented. This handout will instruct you in the use of *elm*. The mail utility *elm* on *bsu-cs* can be invoked by:

```
% elm<return>
```

The first time you invoke *elm*, it will ask you to set up a couple directories in your home directory. Once those directories are created, it will then show a screen similar to the following:

```
Mailbox is '/var/mail/john with 1 message [ELM 2.4 PL11]

1 Oct 21 Henry Markum (19) How's life?

You can use any of the following commands by pressing the first character:

d)elete or u)ndelete mail, m)ail a message, r)eply or f)orward mail, q)uit

To read a message, press . j = move down, k = move up, ? = help

Command:
```

Elm sorts the mail file in the reverse order of time and date and displays the latest mail messages first with the real name and subject line of each message. The current mail message is highlighted. To change the current mail message to another one, simply use the arrow keys to scroll through the list. The paragraph at the bottom of the screen displays the most common commands for *elm*. These commands use the letter before the parenthesis to choose the command. The elm system uses ***vi*** as the default editor for mail messages, but this can be changed by choosing *o (options)* and changing the editor line in the options list.

To send mail to someone at *bsu-cs*, you must only enter the username for the address. To send mail to someone at another site (VAX, another university, etc.) you must enter the host address of the computer where the person has an account. The form for a remote address is as follows:
`remote_username@hostaddress` For example, for a mythical account `john` on the *bsu-cs* system, the entire address would be: `john@bsu-cs.bsu.edu`.

Printer Access

There are many printers that can be accessed from the Unix system. Printers are named by their location (building and room number). For example, the printer in RB 369 is named Prb369, those in RB 355 are

named Prb355. It is usually best to send your printouts to the room in which you are working. The command to print a document in RB369 is: `%lpr -Prb369 filename .`
