平成11年度 国立特殊教育総合研究所 長期研修成果報告書

ロービジョンのHTML教材利用に関する研究

視覚障害教育研究部 氏間 和仁 (愛媛県立松山盲学校)

目 次

1	はじ	ために ····································
	1.1	問題の所在・・・・・・・・・・・・・・・・・・・・・・・1
	1.2	電子化文書の特徴とロービジョンへの応用・・・・・・・2
	1.3	研究の趣旨・・・・・・・・・・・・・・・・・・・・・・・・4
	1.4	研究の流れ・・・・・・・・・・・・・・・・・・・・・・・・ 4
2	HT	ML文書表示ツールの試作 · · · · · · · · · · · · · · · · · · ·
	2.1	表示ツール試作に至るまでの経緯・・・・・・・・・・・・6
	2.2	表示ツールのデザイン・・・・・・・8
	2.3	表示ツールの試作・・・・・・・12
	2.4	ファイル構造の詳細・・・・・・14
	2.5	表示ツールの応用利用・・・・・・・23
	2.6	表示ツールの活用例・・・・・・・24
	2.7	表示ツールの課題・・・・・・・・・・・・・・・・・・27
	2.8	まとめ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・
3	HT	ML教材の有効性に関する実験的検証 · · · · · · · · · · · · · · · · · · 29
	3.1	実験に至までの背景・・・・・・・29
	3.2	仮説30
	3.3	実験の概要・・・・・・・・・・・・・・・・・・・・・・31
	3.4	結果34
	3.5	考察37
	3.6	事例による検討・・・・・・・・・・・・・・・・・・・・・・・・・・・・・40
	3.7	まとめ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・
4	HT	MLの展望 · · · · · · · · · · · · · · · · · · ·
	4.1	HTMLの現状 · · · · · · · · · · · · · · · · · · ·

	4.2	•	アク	セ	シリ	<u>~</u> 1	J ラ	- 1	· ^	\O,)取	しり	組	み	. • •	• •	• •	• •	• •	• •	• •	• •	• •	• •	• •		• •	• •	• •	• •	• •	• •	43	
	4.3]	HTI	ИL	の^	今往	乡							. .		· • •					٠.		• •			• •							45	
	4.4]	HTI	ИL	教	材音	当人	交1.	こま	317	ナる	討	是	1								٠.	• •					• •		٠.		• •	47	
	4.5		まと	め		· • •					• •	• •	• •	• •			. . .									• • •						• •	53	
5	お	おわ	りに	· · ·			• • •	• •	• • •	• • •	• • •		••		· • •		••				٠.		• •	• •				••	• •	••	• • •	· • •	54	
6	謝	辝			••		• • •	••	• • •	• • •	• • •		••			••	•••					٠.	• •			••	• •	••	, .		•••	• • •	55	
7	用	語	集·		••		• •	• •	• • •		• • •		• •		· • •	••		••	••		• •	• •		٠.			• •		• •		• • •	• • •	57	
8	弓	用	・参	考	文詞	猒	• •	• •	• • •	• • •									• •				• •	٠.		••	• •	• •	• :		• •	• • •	63	

..

•

1 はじめに

1.1 問題の所在

盲学校には、大別すると全盲と弱視の生徒が在籍している。全盲の多くは、活字を利用して学習できないため点字や録音図書などを利用して学習している。弱視の多くは、活字を利用して学習している。また、現在教育的な「弱視」のことを、医療で用いる「弱視」との混同を避けるために「ロービジョン(low-vision)」と呼ぶことがある(鳥居、1993)。したがって、本研究でもロービジョンと呼ぶことにする。

現在、学校で用いられている教科書やプリントなどの教材や資料は紙に印刷された、ハードコピーのものが主流である。盲学校でもその現状は同じである。一般的にロービジョンがハードコピーの資料を利用する時には、特殊な眼鏡やルーペ・単眼鏡などの光学機器や、CCTV(Closed Circuit Television; 拡大読書器)・拡大コピーなどの電子機器により、文字を拡大して対応している(大川原ら, 1999)。これらの方法を利用すれば、市販の教科書・雑誌・新聞や通帳などの世にあるハードコピーのほとんどのものを手軽に読むことができる。

また、まぶしさなどの原因から素材の白黒を反転した方が読みやすいロービジョンも存在する(Leggeら, 1985b)。この場合、特殊なコピー機やCCTVを利用すれば、素材そのものを大きくして提供できるだけでなく、白黒を反転して提供することもできる。このようにロービジョン一人一人の読書環境の課題を見つけ、解決し、整備していくことで、これら既存のエイドの恩恵をうけるロービジョンが大勢いることが分かっている(中野ら, 1995)。

しかし、これらの既存の方法で全く問題がないわけではない。例えば、ルーペを利用すると、体幹や頭を上下左右に動かして読まざるを得ないし、これらの光学機器は倍率を高くするほど視野が狭くなり視距離も短くなってしまう。拡大コピーは、拡大するほど用紙が大きくなり、同時に一行が長くなり、縦にも長くなる。その結果扱いにくくなってしまうし、さらに用紙サイズの制限から拡大率にも限界がある。CCTVは読み物をテーブルに乗せて、上下左右のX、Y方向に移動するという操作が必要であり、分厚い本などは操作性に問題がある。また、ロービジョンの中には滑動性眼球運動や衝動性眼球運動の機能が低い者もいることから、それらの眼球運動の機能を調べた上で適切なCCTVの指導も望まれている(葉石, 1995)。さらに、これらの方法で

は1行が長くなってしま うことから、行移りの に行を間違えた とこではページを合ったと さにピントがよがな にない、こまでに時間が経 でないない でいる。

また、これら様々な手 法がある中、どうしても 解決できない問題もある。 それは表 1に示すように、

表 1 代表的なリーディングエイドの機能 中野泰志(1994)より一部を掲載

	144 64-	代表的なリーディングエイド									
	機能	弱視レンズ	拡大テレビ	拡大コピー							
低視力	拡大	0	0	0							
対策	明るさの確保	△(一部機種)	0								
	光量調節	△(一部機種)	0								
羞明	白黒反転	_	0	△(一部機種)							
対策	マスク	_	0	_							
狭視野	縮小/レイアウト表示	△(一部機種)	0	0							
対策	文字間隔変更	_	_								
	アンダーライン	_	△(一部機種)								
中心暗	行間隔変更										
品列果	縦/横書き変更	_									
	色変換		ム(一部機種)	ム(一部機種)							
	オートスキャン	-		-							
i	表示横幅変更	-	△(マスク利用)								
その他	文字列検索	-	_	-							
	ページジャンプ	-	_								
	マーカーセット	_	_	_							
	音声対応	_									

白黒反転や行間隔、文字間隔、1行文字数などが手軽に自由に調整できないことである(中野ら, 1994)。これらの要素は時として、ロービジョンの読書の阻害因子になることが多く報告されている。したがって、これら様々な要素が、ロービジョン一人一人の見え方に適するように設定されることが必要であるといわれている(中野ら, 1994)。

1.2 電子化文書の特徴とロービジョンへの応用

このような問題を解決するために、電子的なリーディングエイドが今までにも作られてきている(佐藤ら, 1993 中野ら, 1994)。このうち中野らの報告は、「文書電子化の際の課題」の中で、SGMLのようなマークアップ言語によるコンテンツの制作が今後大切であると指摘している。

現在、文書の電子化は新聞や書籍、雑誌、各種マニュアルなどにおいて、CD-ROMなどを利用したスタンドアロンによる利用から、インターネットを通じて配信されるものまで盛んに行われている。すでに雑誌メディアと同じ内容がWeb上で見られるものまで登場している(島田ら, 1999)。特にインターネットが普及し始めると、それは分散蓄積され、ハイパリンク化、ハイパメディア化する。この現象を小野(1998)は第2世代の電子化と呼んでいる。

第2世代の電子化の中で、文書の電子化の流れは「データ構造を共有するための形式」と「表示を共有するための形式」の2つの分野で進んでいる(東, 1999)。その一般

的な技法として、データ 構造化共有の分野では SGMLという構造化言語 系のHTMLやXMLが、表 示共有の分野ではAdobe 社が提唱しているPDFが それぞれ普及している。 東は、この2つの流れは、 それぞれ目指すものが違 っているためその善し悪 しを単純には比較できな いと指摘している。

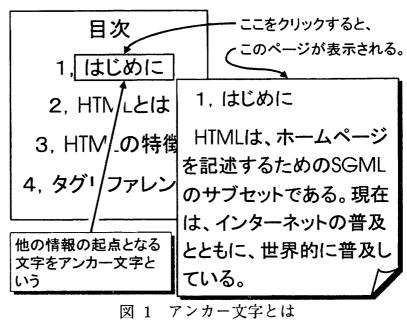


図 1

これらの電子文書メデ

ィアがロービジョンの読書環境を改善する1つの手段として可能性があることは経験 的に分かっている。次に、電子メディアの2つの潮流より、HTMLとPDFがロービジ ョンにもたらす恩恵の可能性についてまとめる。

HTMLで記述された文書の一般的なメリットは、(1)電子文書として流通と管理が 容易になること、(2)動画や音声、シミュレーションプログラムなどマルチメディア文 書が作成できること、(3)ハイパリンクにより文書同士の参照が簡単に実現できること (図 1)、(4)文書の電子化が容易かつ安価に行えること、(5)文書制作に関わるトータル コストの削減に繋がることが期待されることが報告されている(保田、1999)。この中 からロービジョンのメリットを整理すると、(1)ハイパリンクにより文書の移動・検索 がスムーズである、(2)フォントサイズや配色・行間隔や文字間隔などの表示の方法を ユーザーが自由に設定できる、(3)文字を拡大したときに自動的に画面の幅で折り返し てくれるの3つが挙げられる。また、教材としてのメリットは(1)容易で安価に作成で きる、(2)インターネットやCD-ROMで流通させることができることの2つが挙げられ る。

これらのことから、ロービジョンがHTMLにより作成された教材(以下、HTML教 材とする)を用いることにより、ハードコピーの教材とは違った読書上、学習上のメ リットが予想される(氏間、1998)。

一方、製品のマニュアルや論文・パンフレットなどで広く取り入れられ、こちらも インターネットを通じて広く普及しているPDFは、(1)拡大しても書式・体裁が崩れ ない、(2)PostScriptを応用していることから、どんなに文字を大きくしてもきれいで

4 1 はじめに

ある、(3)ハイパリンクにより文書の移動・検索がスムーズであるなどのメリットを持っている。これらのことから、HTML同様に、PDFで記述された教材(以下、PDF教材とする)も、ロービジョンにとってメリットがあると考えられる。

1.3 研究の趣旨

上記のように、HTML教材やPDF教材の電子化教材はロービジョンにとって読書や 学習上のメリットが予想されるにもかかわらず、現在それらに関する研究は見当たら ない。

そこで、本研究ではHTML教材について、ロービジョンが利用しやすい表示ツールの試作を行う。そのツールにより表示されたHTML教材、PDF教材、ハードコピーの教材(以下、ハードコピー教材とする)の3つのメディア間で、特に読書や学習に影響を及ぼすと考えられる、検索時間や読速度にどのような違いが認められるのかを実験により調査する。そして、その結果を視力や視野などの視機能、PC(Personal Computer)使用歴などの視点から分析し、ロービジョンが電子化教材を扱う際の有用性や特性についてハードコピー教材と比較し、確認する。HTMLを中心とする電子文書の現状と今後の動向や課題を概観する。

ロービジョンの眼疾や視力・視野などの視機能の障害の状態は一人一人異なるため、すべてのロービジョンに有効な読教材の条件を設定することは大切なことであるが、同時に困難であるといわれている(大川原ら, 1992)。しかし、前述した通り電子化教材はこの困難さを克服する可能性を秘めている。本研究では、ロービジョン一人一人の読書環境に適応した読み物が手軽に提供できるHTML教材の重要性を示したい。

1.4 研究の流れ

本研究は、以下の3つの項目から構成される。

1.4.1 HTML文書表示ツールの試作

HTML文書のレイアウトを手軽に設定でき、マウス操作を支援するようなHTML文書表示ツールを試作する。その過程でロービジョンの読書の特性について十分に検討し、それを生かして試作したい。また、HTML教材を応用して作られたHTMLの活用例も示し、試用した感想も載せる。

1.4.2 メディアの比較実験

表示ツールを試用したHTML教材の有効性をハードコピー教材、PDF教材と比較す るための実験を行う。3つのメディアについて読速度と検索時間の2つの観点から比較 検討する。

1.4.3 今後の展望

電子文書の現状に触れ、HTMLやXMLなどのSGML系の技術を中心に、電子文書 の今後や課題についてまとめる。

6

2 HTML文書表示ツールの試作

2.1 表示ツール試作に至るまでの経緯

HTML文書は、様々な方法により、文字サイズや配色などのレイアウトの設定を行うことができ、ロービジョンにとってのメリットが予想される。その代表的な方法は次の3通りが考えられる。(1)ブラウザソフトの機能を利用してレイアウトを設定して表示する、(2)JavaScriptやCGIなどによりHTMLを操作して表示する、(3)専用のブラウザソフトによりレイアウトを設定して表示する。次に、これらの方法の特徴について解説する。

2.1.1 ブラウザを設定し表示する

この方法は、大きく分けると(1)ブラウザの設定機能コマンドを用いる方法と、(2)CSSをブラウザに読み込ませて使用する方法の2種類に分けられる。

(1)の方法は、ブラウザ ソフトのメニューの中か ら画面の設定を行うもの である。図 2はブラウザ ソフトでトップシェアの、 Internet Explorer 50 ∨ イアウトの設定画面であ る。ここでは、文字色、 アンカー文字色などの配 色設定、フォントの設定 を行うことができる。し かし、文字間隔や行間隔、 などの細かな設定を行う ことは不可能である。ま た、ショートカットキー を覚えないと、マウスで

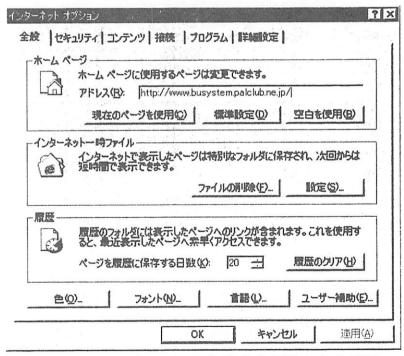


図 2 ブラウザソフトで表示形式を設定する画面

の操作が中心になってしまうため、ロービジョンにとって使いやすい環境とは言い難

1,0

(2)CSSを利用するとかなり細かな書式情報までコントロールできる。CSSの詳細については「4.4.1.1 HTML文書の構造」の項で記すが、HTMLのレイアウト情報を設定する規格である。したがって、それぞれのロービジョンの見え方に適した読書環境を再現する

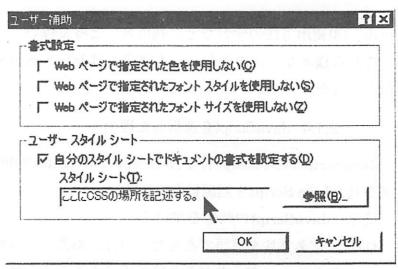


図 3 ユーザー補助のフォーム

ためのレイアウト情報をCSSで記述し、図 2の画面の「ユーザー補助($\underline{\mathbf{E}}$)」のボタンを押すと図 3のフォームが開く。このフォームの「ユーザースタイルシート」の「自分のスタイルシートでドキュメントの書式を設定する($\underline{\mathbf{D}}$)」をチェックする。すると「スタイルシート($\underline{\mathbf{T}}$)」が入力可能状態になるので、図中の黒矢印部分へCSSのファイル名を記述するか、「参照($\underline{\mathbf{B}}$)…」ボタンを押してCSSファイルを選択してCSSファイルを設定する。こうすることで、CSSで記されているレイアウト情報にしたがって、HTML文書を表示させることができる。

しかし、CSSを自分で書くとなると、CSSの勉強をする必要があるので、すぐに使 うということにはならない。

2.1.2 JavaScriptやCGIなどによりHTMLを操作して表示する

この方法は、大きく分けると、(1)CGIなどによりサーバ側でHTMLやCSSを生成して表示する方法と、(2)VBScriptやJavaScriptなどのようにHTMLの中で動作して表示する方法の2種類がある。この方法は、利用者が扱いやすい専用のインターフェイスを作成できるし、インストールの手間も必要ないことから、3つの方法の中では利用者に負担をかけにくい方法であると考えられる。情報提供者はプログラム言語を勉強する必要があるが、利用者がそれを学ぶ必要はない。

2.1.3 表示用のプログラムにより表示する

これは、T-Timeなどの特殊なブラウザソフトによりHTMLの書式を自由に設定して表示する方法である。この方法は、専用のソフトウェアだけあって配色やフォント・文字間隔・行間隔・縦書き横書きなどのように、かなり細かな設定を行い表示することが可能である。しかし、この方法は、(1)ソフトの購入、(2)ソフトのインスト

ール、(3)使用方法の学習など、利用者にとってはいくつかの障壁が生じる。 これら様々な方法の中から、本研究ではJavaScriptを用いて表示ツールを作成する ことにした。

2.1.4 JavaScriptを選択した理由

JavaScriptはNetscape社がNetscape Navigator2.0のベータ版を公開したときに実装されたLive Scriptを起源にしている。

簡単にJavaScriptの特徴を挙げる。

- (1) インタプリタ言語であるため製作にあたって特別なツールが必要なく、単純なテキストエディタがあれば作成できるため安価で手軽である。
- (2) ダウンロードされたあとブラウザ側(利用者側)で動作するため、Webサーバの 管理者でなくても簡単に表示ツールプログラムに手軽に機能の変更や追加など のメンテナンスができる。
- (3) HTMLに埋め込まれて動作するため、WWWサーバ以外にCD-ROMによる配布も可能である。
- (4) VBScriptのように、あるブラウザソフトを限定しない。
- (5) キーボードによる操作が可能である。

次に、JavaScriptの問題点を考えてみる。JavaScriptの短所は、(1)インタプリタ言語であるため、ソースコードが盗まれやすく、動作が遅い、(2)ブラウザソフトに依存する部分があるため、表示するソフトウェア環境により一部の機能が限定される、が挙げられる。

これらの長所と短所から、視覚障害者がPCを操作する上で必須ともいえる、キー操作が可能なこと、インターネットに接続されていない生徒のためにCD-ROMでの配布が可能なこと、安価にできることなどのことを考え合わせると、JavaScriptを用いることは最も適切な選択だと考えられる。これらの理由により、本研究では、JavaScriptを用いてHTML文書表示ツールを制作した。

2.2 表示ツールのデザイン

表示ツールは、次の2点に配慮してデザインし制作する。

- (1) ロービジョンの読書パフォーマンスを阻害する要因を検討して、その変数を操作できるようにする。
 - (2) PCを利用する上で、ロービジョンが不得意と経験的に考えられるマウス操作を

極力減らし、できるだけキーボードで操作できるようにする。または、マウス操作を 行う場合でもロービジョンに配慮した操作環境を考える。

2.2.1 ロービジョンの読書パフォーマンスを規定する要因

ロービジョンの読書パフォーマンスに影響を与える要因は様々である。ここでは、 表示ツールでどの変数を操作するのが望ましいかということを知るために、ロービジョンの読書に関する研究をまとめる。

2.2.1.1 文字サイズ

文字サイズについては、すでに多くの研究が行われている。今日、文字の大きさと読書効率の関係においては、文字が大きすぎても小さすぎても読書効率が悪くなることが知られている(Leggeら, 1985b)。また、それぞれの眼の状態に応じて読書効率が最大になる文字サイズ(CPS; Critical Print Size)があることが指摘され、ロービジョンの読書効率を最大限に引き出すためには、最適な文字サイズを検討する必要のあることが分かってきている(中野ら, 1992)。最近では日本でもCPSを求めるための検査チャート「MNREAD-J」が開発され、臨床で利用され、様々な読書効率に関する研究でもその重要性は指摘されている(小田ら, 1998)。

したがって、文字サイズについては各個人で最適なサイズがあるため、自由に設定 できるようにする。

2.2.1.2 フォント

フォント(書体)についてもいくつかの報告がされている。小田ら(1993)はフォントの見やすさの研究の中で、教科書体と明朝体、ゴシック体の3書体を人工的にほやかしてどのフォントが見やすいかを定量的に測定した。その結果、ゴシック体が最もほやけに強く、明朝体、教科書体の順であった。画数の多い漢字ではフォントによる違いが無く、文字が複雑になるとどの書体でも鮮明にしなければならないと報告している。

したがって、今回のツールではゴシック体に固定して表示するようにした。

2.2.1.3 配色

配色とは、白地に黒文字の一般的な設定と、それを逆転した黒地に白文字の設定のことである。VDT(Video Display Terminal)に関する研究の結果では、健常者の場合、黒文字に白背景の条件の方が作業効率が高いことが分かっている(窪田, 1989)。また、Leggeら(1985a)は晴眼者において、文字と背景の色の極性を反転させると読書効率が

良くなるというコントラストポラリティ効果による影響は少ないと報告している。しかし、下記の通り、ロービジョンの中にはコントラストポラリティ効果を示す者の存在が報告されている。

Leggeら(1987)は、テレビモニターに文字を流して、それが認識できる最大の速度を指標として、黒文字に白背景の条件(B/w)と白文字に黒背景の条件(W/b)での読速度の違いを検討している。その結果、晴眼者と透光体に混濁のないロービジョンでは両方の条件で差は認められなかった。しかし、透光体に混濁のあるロービジョンではB/wの条件の成績が良く、読速度の向上がみられたと報告している。さらに、このようなコントラストポラリティ効果については、古田(1989)らや中野(1991)も同様の報告をしている。

また、最近の研究では、羞明を訴えるロービジョン全員にコントラストポラリティ効果がみられるのではなく、白子症や無虹彩ではW/bとB/wの条件間で読速度に影響を及ぼさないという報告もされている(新井, 1999)。

そこで、本ツールでは、白文字に黒背景(W/b)条件と、黒文字に白背景(B/w)条件を選択できるようにした。

2.2.1.4 文字間隔

文字間隔について、Leggeら(1985b)は、晴眼者2名、ロービジョン4名(2名は透光体 混濁で中心視野に欠損がない者、2名は透光体混濁無しで中心視野欠損がある者)を対 象として、視角6°の文字を文字間隔6°、9°、12°で提示したときの読速度を測定 した。その結果、晴眼者・ロービジョンともに、文字間隔が広くなると読速度が低下 することを報告している。また、文字間隔を狭めたときの要因について、弓削(1966) はロービジョンの中には字詰まり視力が低下する者が存在することを報告している。 最近の研究では、CPSと文字間隔について、字詰まり効果はCPSには影響するが、最 大読速度には影響を与えないこと、また字詰まり効果が読速度に影響するのはCPSと それよりも小さい文字サイズの時であることなどが明らかになっている(川嶋ら、 1998)。

さらに、読書課題を用いて、文字間のスペースの効果が臨界文字サイズを境にして どのように異なるか、中心視野欠損のあるロービジョンの被験者で検討している(川 嶋ら:1999)。その結果、字詰まり効果による影響は、中心視野欠損のあるロービジョ ンの重篤な読書困難をもたらす有力な要因ではないことが導かれ、さらに字詰まり効 果は臨界文字サイズ未満だけで起こる現象であることが報告されている。そして文字 サイズを臨界文字サイズ以上になるように設定すれば字詰まり効果による読速度への 影響はおこらないと結論づけている。

これらの結果から、臨界文字サイズを常に補償できるのであれば、文字間隔は固定でも良いということである。しかしながら、読み物の中には、表のように拡大してしまうとかえって読みづらいと思われるものや、英語のように詰まっている方が読みやすいものもある。教材の提示を目的として表示ツールの開発を行うのであれば、これらの様々な状況に対応できるようにすべきであろう。

そこで、本ツールでは文字間隔の変数も変更できるようにした。

2.2.1.5 行間隔

南地ら(1992)は、読みやすさに及ぼす行間隔の効果の研究を行っている。これによると、行間隔が狭すぎると、隣接する行の影響を受けて読みやすさが低下すると考えられ、その意味では隣接する行が存在しない一行提示の方が読みやすくなるとしている。しかしながら、ロービジョンの場合では、文字サイズの大小に関わらず、多行提示と一行提示とで正読字数が等しくなるような行間隔の条件が存在し、これを境に、行間隔が広くなると多行提示の方が、正読字数が増加するという傾向が見られたと報告している。そして、行間隔にも臨界の行間隔が存在しているとしている。つまり、行間隔をどの程度とるのがロービジョンにとって最も良いというような一般的基準は存在せず、個々のロービジョンによって適切な行間隔を設定することが望ましいといえよう。

したがって、本ツールでは行間隔も任意に設定できるようにした。

2.2.1.6 その他

その他に、読書に与える影響として考えられるものに、1行文字数が挙げられる。これに関する先行研究は菊地ら(1993)が行っている。それによると、ロービジョンの最小可読文字サイズ付近の文字サイズ条件では行間隔の広い条件と一行条件が最も正読字数が多かった。しかし、行間隔が狭い条件においても1行文字数を調節することでパフォーマンスを得ることができる。行間隔の広い条件と一行条件では、正読字数は一行の長さが4~11文字程度の時にピークとなり、行間隔の狭い条件では、正読字数は一行の長さが4~6文字程度の時にピークになるとしている。

しかしながら、最近の臨床や研究では最小可読文字サイズよりも、CPSを重視するようになっており、本研究でもCPSを利用することから、菊地らの研究で得られた一行文字数にはとらわれないようにした。

2.2.1.7 レイアウトの設定のまとめ

以上のことから、ユーザーがカスタマイズできる項目は、文字サイズ・行間隔・文字間隔・配色とし、固定項目は、フォント(ゴシック体)、1行文字数(画面幅)とした。

2.2.2 マウス操作

ロービジョンとマウス操作に関する先行研究は見当たらない。しかし、マウス操作はロービジョンがPCを利用する上で障壁の1つになっていることは経験的に分かっている。したがって、なるべくマウス操作を行わなくても利用できるようにデザインすることにした。

以上のことから、キー操作が取り入れられる部分は、なるべくキー操作が行えるようにし、マウス操作が必要な場合はそれを支援するような工夫をすることにした。

2.3 表示ツールの試作

前述したことが実現されるようにデザインして、表示ツールを試作したので、その 過程と仕組みについて解説する。

2.3.1 試作1版

まず、試作1版を作った。これは、表示部と環境設定部が1つのページで行えるタイプである。つまり、なんらかのHTML文書を見ながら、文字を大きくしたり、白黒反転をしたりすることができる。日本文では文字間隔を広めにとって読んでいて、続いて英文を読むようなときには、その場で読み物を参照しながらレイアウトを変更できるメリットがある。しかし、この方法では、文字サイズを変更するためのボタン・白黒反転を行うためのボタン・行間隔を変更するためのボタンのように変更したい変数の数だけボタンが必要になる。初心者には使いづらいし、実際に、何名かのロービジョンに試してもらったが、不評だったため表示部と環境設定部を分けて開発することにした。

2.3.2 試作2版

試作1版の結果を踏まえて、表示ツール試作2版は、表示部と設定部の2つの部分に 分かれている。

表示部は、1つのJavaScriptプログラムファイル(text.js)と、2つのスタイルシート用ファイル(Wb.cssとBw.css)をホームディレクトリに置き、そのファイルを同階層または、下位階層の全てのHTMLファイルから参照して文字サイズや行間隔・文字間

隔・配色などを再現するようになっている。こうすることでプログラムのメンテナンスが行いやすくなる。また、プログラムファイル(text.js)は、クライアントマシンのcookieに記録されたデータを読み込み、環境設定用の2種類のCSSファイルを選択的に読み込んでHTML文書全体を表示する(図 4)。

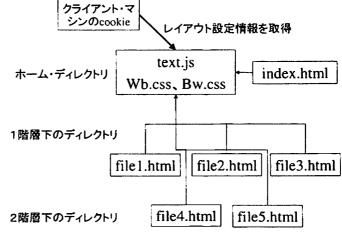


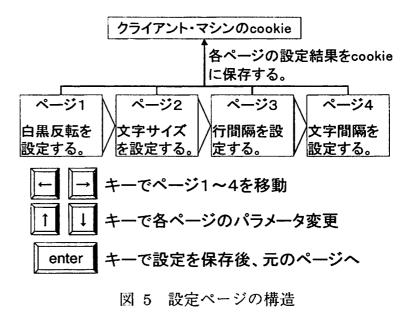
図 4 表示ツールのファイル構造

設定部は、4つのHTMLファ

イルから構成されている。ページ1では白黒反転の設定、ページ2では文字サイズの設定、ページ3では行間隔の設定、ページ4では文字間隔の設定を行うようになっている (図 5)。ここで設定したレイアウト情報は、クライアント側のコンピュータにある cookieに保存され、すべてのHTML文書は表示用プログラム(text.js)によって、この cookieファイルの中から必要なレイアウト設定のデータを呼び出してレイアウトをコントロールするようになっている。

cookieにレイアウトの設定値を保存することにより、一人一人の設定値はクライアントコンピュータに保存されるので、同じHTML教材を複数の利用者が閲覧しても、

それぞれの利用者の設定がれるように1台のはなる。コように1台のはからに1台のはなりに1台のはなりでする。コようのはで使うないのながです。 Windowsへテぞののよのながでも、変えで、それののokie ファインを作る。 アイイがにはないである。 PCを何としたのである。 ではない アイル にしたい アイル が 再現される。



2.4 ファイル構造の詳細

表示ツールについて、ソースコードを参照しながら、動作の原理を詳しく解説する。 ソースコードは巻末に資料として載せた。

2.4.1 表示部の構造

このツールを利用するHTMLファイルのヘッダにはホームディレクトリ上のtext.js ファイル参照用のタグが記述されている必要がある。

ホームディレクトリ用は、

<script language="JavaScript" src="text.js">

</script>

で、1階層下用は

<script language="JavaScript" src="../text.js">

</script>

である。ここで示しているように、text.jsの場所だけは相対パス指定により各HTMLファイルに記述する必要がある。2階層下なら、src="../../text.js"、3階層下なら、src="../../text.js"となる。このタグを各HTMLファイルのヘッダに埋め込むことにより、ホームディレクトリのtext.jsファイルが、まるでそれぞれのHTMLファイルに記述されているプログラムであるかのように動作するのである。ただし、ここでは流通のことを考えて相対パス指定にしてあるが、text.jsが置かれているURLが決まっている場合は絶対パス指定の方がメンテナンスは簡単である。

2.4.1.1 HTMLコンテンツを表示する過程

表示ツールを組み込んだページにアクセスすると、13行目がcookieの情報を参照して、それが初回の訪問か、2回目以降の訪問かを判断し分岐するようになっている。その結果初回の表示の場合は、以下の15行目から27行目のルーチンが実行される。

13行目	N=getCookie("Ns");	
14行目	<i>if</i> (N==''''){	
15行目	N=30;	<i>川フォントサイズ用</i>
16行目	FC="white";	// <i>文字色用</i>
17行目	BC="black";	//背景色用
18行目	LC="white";	//リンクテキスト用
19行目	VC="white";	//訪問済みリンクテキスト用

```
リアクティブ文字色
20行目
               AC="white";
                                //行間隔設定
21行目
               LH=15;
                                //文字間隔設定
22行目
               LS=5;
                                //URL格納
23行目
               UL=document.URL;
24行目
               varA=UL.length;
25行目
               varA=varA-10;
26行目
               UL=UL.substring(0,varA);
27行目
               Save();
28行目
               }
```

ここでは、文字サイズは30ポイント、文字色は白、背景色は黒、その他の文字色は白、行間隔は1.5行、文字間隔は1.5文字にレイアウトの初期値として設定される。さらに24行目から27行目では、このツールが置かれている場所をcookieに絶対パスで保存するようになっている。この絶対パスの情報にもとづき、どの階層からでもCSSファイルを読み込んだり、環境設定ファイルへ移動したりできるのである。なお、25行目で絶対パスの文字長から10を減じる操作をしている。これはホームページに訪れるときに必ずindex.htmlファイルが開かれることから、その文字数を引いて、パスのみをcookieに格納するようにしてある。それらの処理の後、27行目でSave関数を呼び出している。

Save関数は、次のの11行からなっている。

```
function Save(){
101行目
102行目
                        setCookie("Ns",N);
103行目
                        setCookie("FCs",FC);
                        setCookie("BCs",BC);
104行目
                        setCookie("LCs",LC);
105行目
                        setCookie("VCs", VC);
106行目
                        setCookie("ACs",AC);
107行目
108行目
                        setCookie("LHs",LH);
109行目
                        setCookie("LSs",LS);
110行目
                        setCookie("ULs",UL);
111行目
```

このSave関数で行っていることは、それぞれのレイアウトの変数名とその値を setCookie関数へ引き渡して、cookieへ保存する作業である。cookieへ直接的に書き込むのはこのsetCookie関数である。その内容は、

の5行である。setCookie関数は2つの引数を受け取って動作する。2つの引数とは、変数名とその値である。95行目のkeyとvalがそれにあたる。したがって102行目のsetCookie("Ns",N);から呼び出された場合を例にとると、Nsという変数名とNに格納された値の両方をsetCookie関数へ引き渡していることが分かる。96行目・97行目で受け取った引数からcookieへの保存に適した形に整形し、98行目でオブジェクトを利用してcookieに保存している。

2回目以降に訪れて表示する場合は13,14行目から分岐して、以下のルーチンが動作する。

```
45行目
          function Effect(){
                                      川フォントサイズ用
46行目
               N=getCookie("Ns");
                                      //文字色用
               FC=getCookie("FCs");
47行目
                                      //背景色用
48行目
               BC=getCookie("BCs");
                                      //リンクテキスト用
49行目
               LC=getCookie("LCs");
                                      //訪問済みテキスト用
50行目
               VC=getCookie("VCs");
                                      //アクティブ文字色
51行目
               AC=getCookie("ACs");
                                      //行間隔
52行目
               LH=getCookie("LHs");
                                      //文字間隔
53行目
               LS=getCookie("LSs");
54行目
               LH=N*(LH/10);
55行目
               LS=LS/10;
56行目
          1
```

46行目から53行目までは、レイアウトを設定するための値を格納するための変数が左辺に、cookieからその変数を読み出す関数が右辺に書かれてあり、//の後ろは、それぞれの変数の説明でプログラム実行中は無視される。54行目のLHは行間隔の値が格納されている。ここで、N*(LH/10)の計算を行い、改めてLHに代入している。これは、cookieには小数が格納できないため、1.5行のような少数をcookieに保存する場合はその値を10倍しておくため、この計算式ではまずLHの値を10で割って、それを文字サイズの比に直すためにNを乗じている。55行目は文字間隔の変数であるが、これも行間隔と同じ理由により10で割っている。

この表示ツールにはWb.cssとBw.cssの2つのCSSファイルを選択的に利用してHTMLコンテンツのレイアウトをコントロールしている。以下のコードがその分岐の部分である。

```
34行目
            if (FC=="black") {
35行目
            sCSS = '<LINK REL="stylesheet" TYPE="text/css" HREF="' +
            UL + 'Bw.css'' > ';
36行目
                 }
37行目
            else {
38行目
                 if(FC=="white"){
39行目
            sCSS = '<LINK REL="stylesheet" TYPE="text/css" HREF="' +
            UL + 'Wb.css''>';
40行目
                 }
41行目
```

Wb.cssを利用するか、Bw.cssを利用するかを決めるのは、FC変数(文字の色を格納する変数)に格納された値である。文字の色が黒ならばBw.cssファイルを、それ以外で文字色が白ならWb.cssを、それぞれ利用してHTMLコンテンツを表示するようになっている。このときに、Wb.cssやBw.cssのファイルは、UL変数へ格納された絶対パスにもとづいて読み込まれるようになっている。

以上のような流れの後、HTMLコンテンツを表示する機能が働く。各HTMLファイル中のタグの1つに、

<body id="L1" onload="Effect()" onkeydown="KeyC()" ondblclick="Back()"> というタグがある。この中の、onload="Effect()"という命令により、HTMLファイルが読み込まれる課程で、text.jsのEffect関数が呼び出され最終的にHTMLコンテンツのレイアウトをコントロールしている。Effect関数は以下の35行からなっている。

45行目	function Effect(){	
46行目	N=getCookie("Ns");	//フォントサイズ用
47行目	FC=getCookie("FCs");	// <i>文字色用</i>
48行目	BC=getCookie("BCs");	//背景色用
49行目	LC=getCookie("LCs");	//リンクテキスト用
50行目	VC=getCookie("VCs");	//訪問済みキスト用
51行目	AC = getCookie("ACs");	//アクティブ文字色
52行目	LH=getCookie("LHs");	//行間隔
53行目	LS=getCookie("LSs");	//文字間隔

84行目

```
54行目
                  LH=N*(LH/10);
 55行目
                  LS=LS/10;
 56行目
 57行目
 58行目
                if(document.layers){
                  document.layers["L1"].style.letterSpacing=LS+"em";
 59行目
 60行目
                  document.layers["L1"].style.lineHeight=LH+"pt";
                  document.layers["L1"].style.fontSize=N+"pt";
 61行目
 62行目
                  document.bgColor=BC;
 63行目
                  document.fgColor=FC;
 64行目
                  document.linkColor=LC;
 65行目
                  document.vlinkColor=VC;
                  document.alinkColor=AC;
 66行目
 67行目
                }
 68行目
 69行目
                if(document.all){
                  document.all.L1.style.letterSpacing=LS+"em";
 70行目
 71行目
                  document.all.L1.style.lineHeight=LH+"pt";
                  document.all.L1.style.fontSize=N+"pt";
 72行目
 73行目
                  document.bgColor=BC;
                  document.fgColor=FC;
 74行目
 75行目
                  document.linkColor=LC;
 76行目
                  document.vlinkColor=VC;
 77行目
                  document.alinkColor=AC;
 78行目
                }
 79行目
まずEffect関数が行うことは、cookieに格納されたレイアウト用の変数を呼び出す作
業である。それが46行目から53行目である。ここで使用される関数がgetCookieであ
る。getCookie関数は、
 81行目
             function getCookie(key){
 82行目
                  tmp=document.cookie+";";
 83行目
                  tmp1=tmp.indexOf(key,0);
```

if(*tmp1!=-1*){

```
85行目
                         tmp=tmp.substring(tmp1,tmp.length);
86行目
                         start=tmp.indexOf("=",0)+1;
87行目
                         end=tmp.indexOf(";",start);
                  return(unescape(tmp.substring(start,end)));
88行目
89行目
                  else{
90行目
91行目
                         return("");
92行目
                  }
93行目
            }
```

である。getCookie関数は引数を1つ受けて動作するようになっている。引数はレイアウト用の変数を呼び出すためのキーワードである。46行目を例に取るとN=getCookie("Ns");のようにgetCookie関数にNsという引数を渡しているのが分かる。これを受けてgetCookie関数はクライアントマシンのハードディスク中のcookieファイルからNsというキーワードを元に検索し、それに該当するデータがあれば、その値を返してくれるのである。Nsは文字サイズを表すキーワードであるので、46行目のgetCookie("Ns")により文字サイズが返されて、左辺のN変数へ文字サイズの値が代入されるのである。このような過程をたどって、FCsにより文字色が、BCsにより背景色がcookieより引き出され、それぞれFCやBC変数に代入される。

この一連の変数の呼び出し作業が終わると、最終的な表示作業に入る。それが58行目から78行目である。59行目から66行目と、70行目から77行目はほぼ同じ働きをしている。はじめの8行がNetscape用のルーチンで、後の8行がInternet Explorer用のルーチンであり、それぞれ58行目と69行目で分岐するようになっている。しかし、現在の所、Netscapeの方は上手く動作していない。54行目と55行目は前述の通り、行間隔と文字間隔をそれぞれ小数に直し文字サイズを元に値を設定する部分である。

2.4.1.2 マウス操作の仕組み

このツールで提供しているマウス操作の環境は、(1)ダブルクリックで前のページへ戻る、(2)アンカー文字にマウスポインタが重なると反転して知らせてくれるの2つである。

はじめに、ダブルクリックで前のページへ戻る機能について説明する。マウスがダ ブルクリックされたことを検出するのは下記のタグである。

<body id="L1" onload="Effect()" onkeydown="KeyC()" ondblclick="Back()">

これは各HTMLファイルに記述されているタグの1つである。この中のondblclickの

部分がマウスのダブルクリックを検出し、Back関数を呼び出すようになっている。 Back関数は次の3行からなる小さな関数である。

124行目 function Back(){

125行目 history.back()

126行目 }

ここで、用いられているオブジェクトはhistory.back()である。このオブジェクトはブラウザの履歴をさかのぼって表示させるときに用いる。一連の動作をまとめると、

HTMLコンテンツを表示中にそのエリア上でマウスをダブルクリックすると、それが <body>タグに書かれたイベントハンドラにより検出され、Back関数を呼び出し、Back 関数では、履歴を1つさかのぼって表示するという流れになる。

次にマウスポインタがアンカー文字上に重なっているときに文字が反転する機能について説明する。この機能はCSSファイルにより実現される。それぞれのCSSファイルには次のようにセレクタが設定してある。

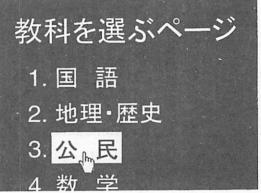


図 6:アンカーにマウスが重なったと きの状態(白文字に黒背景の環境 で)

Bw.cssファイル

A:hover {color:white;

background-color:black;

}

Wb.cssファイル

A:hover {color: black;

background-color:white;

Aというのはアンカータグ用のセレクタである。hoverはこのセレクタの疑似クラスである。図 6で示すようにアンカー文字上にマウスポインタが重なると、この疑似クラスに設定されたスタイルが適用される。この疑似クラスは、Bw.cssファイル(黒文字に白背景の時に呼び出されるCSSファイル)では、文字が白で背景が黒になるように設定してある。これによって見かけ上は反転して見えるようになる。Wb.cssファイ

ルを利用して表示するときも同じ理由で反転して見える。

アンカーセレクタにはfocusという疑似クラスもW3Cでは定義されている。これは TABキーでアンカー文字をセレクトしたときに動作する。これが利用できれば、マウスを使わなくてもTABキーでアンカー文字をスキャンすることができる。しかし、現在のブラウザソフトがこれに対応していないため現在はその恩恵を受けることができない。

2.4.1.3 その他の機能

スペースキーで環境設定へ移動する機能 この機能を働かせるために以下のようなタグをHTMLファイルに埋め込む。

<body id="L1" onload="Effect()" onkeydown="KeyC()" ondblclick="Back()">

このタグの働きは、キーボードのキーが押されると、onkeydownによりそれが検出されKeyC関数を呼び出すことである。

```
117行目 function KeyC(){
118行目 KeyPushed=(event.keyCode);
119行目 if(KeyPushed=="32"){
120行目 OpenWin();
121行目 }
```

KeyC関数は117行目から122行目までである。この関数の機能はキーボードで押されたキーの種類をそのコードにより検出して、それに応じて様々な機能を呼び出すことである。119行目の32という数字がスペースキーを表すコードである。現在は「スペースキーが押されたら、OpenWin関数を呼び出す」という1つの機能しかないが、このように関数を分けておくと、後で機能を追加しやすい。

```
113行目 function OpenWin(){
114行目 location.href = UL + "config1.html";
115行目 }
```

KeyC関数から呼び出されるOpenWin関数は113行目から115行目である。この関数の機能は、環境設定ファイルの1ページ目にあたる、config1.htmlを開くことである。ここでも先にcookieに保存した絶対パスが利用されている。そのパスはULという変数に代入されているため、その内容は見えないが、UL+"config1.html"というのが、環境設定ファイルのURLを示していることになる。ULにhttp://www.******.ne.jp/usr/というURLが代入されているとすると、

http://www.*****.ne.jp/usr/config1.htmlという具合に環境設定ファイルをURLで絶対パスとして指定できる。この機能により、どの階層からでも環境設定ファイルへ移動することが可能になっている。

図を反転させて表示する機能

この機能はWb.css(白字に黒背景の時に使用されるCSSファイル)に以下のセレクタを記述して実現される。これによりタグで表示されるイメージファイルをネガポジ反転させて表示することができる。

Img {filter:invert()}

ただし、このセレクタは現在Internet Explorer5.xでしか対応されていないため、この機能を使用する段階でブラウザの種類が限定されてしまうところが問題である。

2.4.2 レイアウト設定部分の構造

レイアウト設定部では、白黒反転・文字サイズ・行間隔・文字間隔を4ページで調整できるようにした。ファイルは、config1.htmlからconfig4.htmlまでの4つで構成される。

まず、環境設定ページへ移動するには、表示ページでスペースキーを押す。

ページ1では、配色が設定できる。下カーソルキーで白文字に黒背景、上カーソルキーで黒文字に白背景に設定できる。

ページ2では、文字サイズが設定できる。上カーソルキーで文字を大きく、下カーソルキーで文字を小さくできる。文字サイズを6ポイントから100ポイントまで、1ポイント単位で設定できる。

ページ3では、行間隔を設定できる。上カーソルキーで行間隔を広く、下カーソルキーで行間隔を狭くできる。行間隔0から2まで、0.5行間隔で設定できる。

ページ4では、文字間隔の設定ができる。上カーソルキーで文字間隔を広く、下カーソルキーで文字間隔を狭くできる。文字間隔0から2文字まで0.5文字間隔で設定できる。

各ページの移動は、左右のカーソルキーで行える。どの設定ページからでも、エンターキーを押すことで、環境設定部に移動する前の表示ページに戻ることができる。 cookieへの保存は、左右カーソルキーを押したとき、つまり、各設定ページ間の移動時と、エンターキーを押して、元の表示ページに戻るときに行われる。したがって、文字サイズを設定して、行間隔を設定した後、その行間隔を確認しながら再び文字サイズを設定できるなど、常に、全ての表示された状態を確認しながら設定することが

できる。

2.5 表示ツールの応用利用

これまでは、表示ツールの基本構造について詳解した。ここでは、この表示ツールを教育現場でどのように利用できるかを具体的に解説する。表示ツールの利用は単にHTML文書を画面上で効果的に見せることだけにとどまらず、以下のような応用利用が考えられる。

2.5.1 個々のロービジョンに対応したハードコピー教材の印刷

現在、ハードコピー教材をワープロ機やワープロソフトで作成する場合、用紙サイズやフォントサイズなどのある一定の書式に基づいて作成されることが多い。そこで作成されたプリント教材を拡大して用いる場合、コピー機でA4からB4やA3のように拡大して用いられることが多いだろう。しかし、用紙のサイズが大きくなると、1行が長くなり読書時には頭の左右の運動が増し、縦にも長くなるので教材の上の方を読むのが困難だったりすることは経験上わかっている。したがって、文字サイズや行間隔を調整して用紙サイズはA4やB5などの手軽に扱える大きさに印刷できると先のような不便もなく、持ち運びなどの取り扱い上も便利である。

しかし、このことをワープロで行おうとすると、文字サイズを変更し、文字サイズに合わせて1行文字数や行数を設定するなどの、書式に関する設定の調整が煩雑になることが予想される。用意する資料が2種類程度ならそれも現実的であろうが、4種類、5種類と増えてくるとこの方法は現実的で無くなる。また、同じ教材を複数のファイル名で管理する必要も出てくるし、さらには、教材の修正が必要なときには全てのファイルを開いて修正することになってしまう。

しかし、この表示ツールを用いれば、文字サイズ・行間隔・文字間隔が容易に設定でき、それをそのまま紙に印刷することで、手軽に各種書式のプリント教材を作ることが可能になる。さらに、ワープロでは難しい、黒地に白文字の資料も簡単に印刷することができる。このように本表示ツールは、単にインターネットやPCを利用する生徒のみだけでなく、ハードコピー教材を中心に学習する生徒にとっても利用できることが分かる。また、教材の管理の問題からも、同じ内容の教材で書式の違うファイルをいくつも管理するよりは、HTML教材ファイルを1つ用意し、出力時に書式を手軽に変更して印刷する方が効率的だと考えられる。

^{*} 画像のリバース印刷は現在対応していない。

2.5.2 音声による利用

本表示ツールは、ロービジョンを考慮して作られている。しかしHTMLファイルそのものは単純なタグのみで作り上げており、JavaScriptプログラムは別ファイルになっている。したがって、ホームページリーダやVoice Explorerなどの音声ブラウザで表示しても、問題なく扱えることが確認されている。

2.6 表示ツールの活用例

本表示ツールの開発にあわせて、いくつかの教科の教材を制作した。その中から日本史と経絡経穴概論の教材を取り上げて表示ツールとHTML教材の活用について解説する。

2.6.1 日本史の例

日本史の中から、「原始・古代」の単元の教材をHTML化した。担当教員の希望は、「キーワードは隠れていて、マウスを重ねることで、キーワードが表示できるようにしてほしい」ということであった。

そこで、CSS中にアンカーセレクタにans(answerの頭文字)というIDを持たせて対応することにした。具体的な記述例は以下の通りである。

Bw.css

```
91行目
            a#ans{
                        color:white;
92行目
                  background-color: white;
93行目
94行目
            a#ans:hover{color:black;
95行目
                 background-color:white;
96行目
                 }
                             Wb.css
96行目
            a#ans{
                        color:black;
97行目
                  background-color:black;
98行目
                  }
99行目
            a#ans:hover{color:white;
100行目
                  background-color:black;
101行目
                 }
```

これらの、CSSを用いて、実際のHTML教材のファイルには以下のように記載する。

<h3>1 文化のはじまり</h3>

<h4>(1)日本列島と日本人</h4>

^①(更新世) = 洪積世
こうすることで、「更新世」の部分が、普段は隠れているが、マウスが重なることにより、現れるようになる(図 7)。これは、本来のタグの用い方からはずれるため、あまり推奨できる方法とはいえない。しかし、現在のところ、疑似クラスがアンカーの

セレクタにしか対応していないため、やむを得ずこの方法を用いた。上記のソースコードに記載の通り、アンカータグは のように記してある。<a>はアンカータグ本体で、その中の「id="ans"」によりCSSに記してあるIDと同期している。つまり、IDにより識別されて配色の設定が行われるのである。アンカー先のファイルは、自身のファイル名を記してある。こうすることで、誤ってクリックしても、HTML教材上で迷子にならないようになっている。

この他に、<form>タグを用いた課題も 制作した。具体的な記述方法は以下のよ うである。

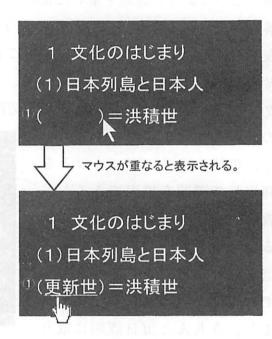


図 7 日本史のHTML教材(マウスが空欄に重なると答えが現れる)

<form action="mailto:ujiman@dokidoki.ne.jp" method="post">
<input type="hidden" name="単元" value="日本史、原始・古代、古代国家の起源、文化の始まり">

名前:<input type="text" size="10" name="Name">(必ず入力)
<hr>

<h1>第1部 原始·古代</h1>

<h2>第1章 古代国家の起源</h2>

<h3>1 文化のはじまり</h3>

<h4>(1)日本列島と日本人</h4>

^①<input type="text" size="6" name="1-1-1-①">=洪積世

<0l>

| (1) 時期 · · 約 < sup > ② < / sup > < input id="i" type="text" size="6" name="1-1-1-2" > 年前~約 < sup >

このように記述することで、図 8のように空欄ができ、そこへ解答が入力できるようになっている。<form>タグのはじまりは、

<form action = "mailto:ujiman@dokidoki.ne.jp" method = "post">

のようである。この「mailto」の部分に各担当教員の電子メールアドレスを明示しておくことにより、課題を解いた後に、送信ボタンを押すと、担当教員へ電子メールで解答が送られるようになる。このような<form>タグを用いたHTMLコンテンツでも表示ツールにより各変数を設定することができるため、かなり幅広い内容のHTML教材の制作が可能であることが分かる。また、1校の校区が広い盲学校にとって、インターネットを利用したこのような双方向の教材を提供できることは、郵便による教材

の配布と回収を行わなくてもよく、生徒は、インターネるし、 利用して課題を提出できるられる環境にさえいれば、学校いれば、学校いれば、学校いればでも、 信を取りに行かなくてもけ取り でも生徒からの解答を受け取り るし、さんと毎日課題にども 組んでいるか、進捗状況なども 知ることができる。

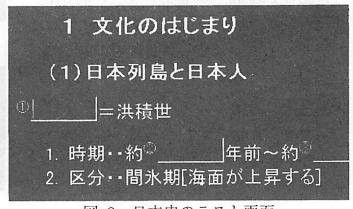


図8 日本史のテスト画面

2.6.2 経絡経穴概論の例

理療科の科目の中に「経絡経穴概論」がある。これは、全身に400種類近くある経穴という治療点について学ぶ科目である。この科目を学習する上での最大のニーズは経穴のデータベースを利用することである。これまでにも、著者はデータベースソフトを利用した経穴データベースを開発しているが、これはデータベースソフトが必要であり、さらにソフトの操作が難解なため、誰でも利用できるものではなかった。そこで、今回は経穴データベースをHTML教材で実現している。データベースといっても、ハイパリンクを多用しただけで、CGIやJavaScriptなどの技術を利用しているわけではない。HTMLを駆使することで十分実用に耐えるデータベースの一例といえる。

図 9は、経絡経穴概論データベースの五十音検索画面である。「あ」「い」「う」な

ど、それぞれの文字をクリックする ことで、その音から始まる経穴一覧 が表示され、その中から表示させた い経穴名をクリックすることで、経 穴名、WHOコード、解剖学的部位、 要穴などの、経穴の詳細情報が表示 される。同じように、経絡別の索引 も用意されており、利用者は好きな

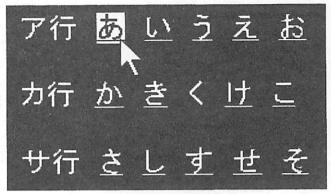


図 9 五十音索引画面

検索方法を選んで使用することができるようになっている。

この経絡経穴概論データベースを理療科の生徒のPCへコピーして試用してもらった。 その結果から象徴的なものを紹介する。

網膜色素変性症、視力0.06、 差明ありの男性生徒は、「白黒反転して非常に読みや すい。CCTVでも白黒反転はできるが、HTMLの方は文字が画面幅内で折り返してく れるので、縦のスクロールのみで良く、使いやすい。またハイパリンクで索引からす ぐに該当の経穴を表示してくれるので、ページ数を確かめたりすることもなく非常に 使いやすい。|と答えている。

わりにしている。画数の多い文字はその場で環境設定画面へ移動して文字サイズを大 きくできるのでとても便利である。早く他の科目、教科以外の内容でも作って欲し い。」と答えている。

しかし、白黒反転がちらちらして見づらい、文字を小さくしたときに行幅を狭くし てほしい、全体的に白色がまぶしいなどの意見も出ており。これらについては今後の 課題である。

2.7 表示ツールの課題

本表示ツールは、前述のように様々な利点があり、応用的な利用も可能であること が分かってきた。次に表示ツールの課題を挙げる。

- 1. Windows用Internet Explorer5.xでのみ全機能の動作確認がとれた。
- 2. ネガポジ反転して画像を表示しても、反転して印刷されない。
- 3. さらに細かな変数の設定が必要。
- 4. 既存のHTMLファイルを、そのまま利用できない。

今後の開発の中で、これらの課題について改善していきたい。

2.8 まとめ

本章では、HTMLコンテンツをロービジョンが読みやすいように自由にレイアウトを設定するための表示ツールの開発とその構造、応用利用や活用について検討した。 その結果以下のことが分かった。

- 1. ロービジョンの読書阻害因子である、文字サイズ・文字間隔・行間隔・配色を自由に設定できる機能が実現された。
- 2. レイアウト設定画面においてカーソルキーのみの操作が実現された。
- 3. マウスの操作性を高めるために、アンカー文字の反転機能、ダブルクリックに よるページをさかのぼる機能が実現された。
- 4. 画像の反転表示機能が実現された。
- 5. HTML教材をブラウザで印刷することにより、ハードコピーを利用する生徒に とっても、利用しやすい教材が印刷できることが分かった。
- 6. 表示ツールを搭載したHTML教材を音声ブラウザで利用できることが確認できた。
- 7. 双方向の課題教材やキーワードをマスキングする教材などの作成が可能であることが分かった。
- 8. 簡単なデータベースを制作することが可能であることが分かった。
- 9. 一部のブラウザでしか全機能が利用できないことが分かった。

この表示ツールを利用したHTML教材が多く現れ、ロービジョンの読書、学習環境が整備されることが望まれる。

3 HTML教材の有効性に関する実験的検証

3.1 実験に至までの背景

ロービジョンと読書効率に関する研究は、ハードコピーを中心に行われている。読書の研究で電子メディアを実験の手段として用いたものは多数行われているが、電子メディアの読書とロービジョンに関する研究は限られている。中野ら(1993 1994)はリーディングエイドをPCで試作している。このエイドは必要性に応じて、文字サイズ、文字間隔、行間隔、1行文字数、縦書き横書き、配色などの文章の提示条件を総合的に変更でき、かなり優れたリーディングエイドである。このエイドを3名の小学生に試用した結果、読書時間の延長や読書へ興味を示す言葉の表出など、どのケースも良い結果になっている。この事例は、PCを用いて読書環境を総合的に調整することが有効であることを示している。また、同研究の課題として文書の構造化が必要であることが指摘されている。そこでHTMLのような構造化言語の必要性が認められるのである。

インターネットの普及に伴って、HTMLやPDFが普及して久しいが、これらの電子メディアを用いたロービジョンの読書効率について確かめた研究は見当たらない。また、中野の研究のように、普段の学習状態と比較して考察を加えたケースはみられるが、同じ読材料を用いてハードコピーなどの既存の読書材料と電子メディアを比較したものも見当たらない。

そこで、著者は試作した表示ツールを用いたHTML教材とハードコピー教材、PDF 教材の3つを比較した実験を行い、電子メディアの有効性を明らかにすることにした。

HTML、ハードコピー、PDFの機能の比較は表 2に示す。表示ツールを用いた

HTML文書は、画面幅で行を折り返し、アンカー文字にマウスが重なると反転して知らせてくれる、理想的な環境であると考えられる。

ハードコピーはハイパ リンクによるページ移動

表 2 メディアの機能の比較

	HTML	HC	PDF
ネットワークによる利用	0	×	0
文字サイズの変更	0	0	0
フォントの変更	0	×	×
行間隔・文字間隔の変更	0	×	×
配色の変更	0	Δ	×
ハイパリンク	0	×	0
行末の折り返し	0	×	×

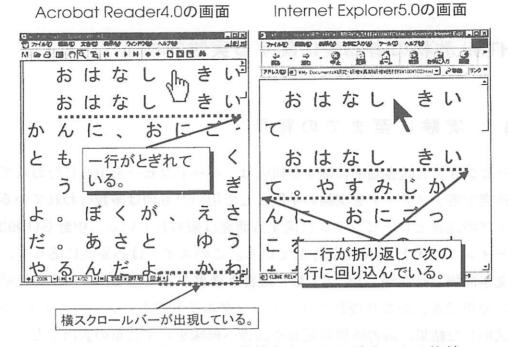


図 10 PDFとHTMLの行替えとマウスポインタの比較

ができないし、拡大すると行幅も広がり、用紙サイズも大きくなり扱いにくくなる。 しかし、多くの人にとってハードコピーは長年扱っているため、慣れているメディア である。したがって、ハードコピー教材と同じ程度のパフォーマンスが電子メディア でも確認できれば、電子メディアは有効な手段であると考えられる。

PDFはアンカー文字の反転を行うことができないためHTMLよりは操作性が悪いことが考えられる。また、PDFは1行文字数や行数などの書式は保持されるため、小さい文字の時は、1行が適切に折り返されるため、画面いっぱいに文字が表示されるHTMLよりは読みやすいことが予想される。しかし、文字を大きくしたときは、図10に示すように画面から1行がはみ出してしまい横のスクロールが必要になるため、パフォーマンスが低下することが予想される。また、HTMLの場合、マウスポインタを背景色と反転して表示できるのに対し、PDFでは白色の指状のマウスポインタに限定されてしまう(図 10)。今回は、背景色を白に固定して実施することから、このこともPDFでの読書パフォーマンスを低下させる要因と考えられる。

3.2 仮説

本実験では、HMTL、ハードコピー、PDFの3つのメディアの違いを読速度と検索時間を用いて比較する。ハードコピーと比べて、HTML、PDFの読速度や検索時間の低下が見られなければ、これらの電子化教材は実用に耐えると考えられる。特に読速

度は、CPSを補償していることから、メディア間での相違が無いことも考えられる。 しかし、PDFにおいて文字が画面からはみ出すほどの大きさになると、ハードコピー やHTMLが速くなることが予想される。また、HTMLは表示ツールを用いることから、 PDFよりも検索時間が短縮されることが予想される。

今回の実験計画では、各メディアの全体の特徴をとらえることを主眼としており、 3つのメディア間に何らかの違いがみられた場合、それがどの要因に起因するのかま で言及することは意図していない。

3.3 実験の概要

3.3.1 目的

HTML教材、ハードコピー教材、PDF教材を使用して、読速度、検索時間を測定し、 それぞれのメディア間にどのような差が見られるか、また、視機能、PC熟練度など によりどのような違いが見られるか調べる。

3.3.2 方法

対象:愛媛県立松山盲学校高等部本科保健理療科1年生、専攻科理療科1・2年生の ロービジョン18名(年齢28±11)

期間:1999年9月·10月

刺激:実験の刺激として用いる読材料の構成は次のとおりである(図 11)。

- 目次
- 1~10ページ:小学校1年生の国語の教科書から 抜出した文章を各ページに書き、各ページの最 後に、次の検索課題が「次は、ももたろうを読 みなさい」のように書いてある。
- 11~20ページ:小学校2年生の国語教科書から 抜き出した文章を各ページに書き、各ページの 最後に、次の検索課題が「次は、ももたろうを 読みなさい」のように書いてある。
- 21~30ページ:小学校3年生の国語教科書から 抜き出した文章を各ページに書き、各ページの 最後に、次の検索課題が「次は、ももたろうを 読みなさい」のように書いてある。

1ページ中の文字数は100字を越えて最初の句点

文章30 小学校3年の 文章 2 文章 1 小学校1年の 教科書より

図 11 刺激の構成

(。)までとする。文字数には句読点やカギ括弧などの記号はカウントしないことにし

た。また、漢字の使用頻度は、課題1は9.69%、課題2は9.78%、 課題3は9.31%のようにほぼ同じ に設定した。

表 3:課題の構成

		H	題順	近二月	んだく	ページ	番号		
課題1	16	1	25	22	14	8	11	6	27
課題2	7	12	21	18	28_	3	19	23_	5
課題3	24	9	13	2	29	15	4	26	17

表 3に示したとおり、1課題の設題は、9タイトル設定し、各学年から3タイトルずつ出題した。また、課題による偏りを相殺するために、課題と3種類のメディアは被験者により組み替えて実施した。

刺激の条件は、以下の通りである。

文字サイズ:各被験者のCPSに設定する。(ハードコピー、PDFは22ポイントを基準にする)

フォント:ゴシック体に設定する。

配色:白地に黒文字に設定する。

文字間隔:0.5文字に設定する。

行間隔:0.5行に設定する。

1行文字数:14文字(ハードコピー、PDF)、画面の幅一杯に表示(HTML)

輝度:ハードコピーの輝度の条件は、文字が6.46cd/m²、背景が79.27cd/m²、コントラストが84.9%、電子化教材の輝度の条件は、文字が20.25cd/m²、背景が99.37cd/m²、コントラストが66.1%であった。金城ら(1989)の報告よりロービジョンにおいて十分なコントラストが保たれているといえる。

解像度:ディスプレイは17inchで1024pix×768pixに設定した。

マウス:マウスポインタは、黒色の最大マウスを用いた。マウスはインテリマウス を用いた。

3.3.3 記録

実験の様子は、MDレコーダで記録した。それを聞きながらストップウォッチで計時した。計時は、課題を読み上げている時間と、読んでいない時間をそれぞれ計り、それぞれ読書時間と検索時間とした。

読書時間は、正読字数を1分間に読む文字数(CPM; Character/Minutes)へ換算した。 9タイトルのCPMを平均し、その課題の読速度とした。

検索時間は、9タイトルの時間を平均し、その課題の検索時間とした。

年 性		性	0P.#=	羞	———— 視	 カ	損失	率	₩	検索		(s)	読返	度(CF	PM)	— CPS MR		PC	
ID	齢	別	眼疾	明	(遠方)	(近見)	(左)	(右)	 	нтм	НС	PDF	НТМ	НС	PDF	<u> </u>	IVING	使用	
AA	24	m	視神経萎縮	±	0.04(0.2)	0.7	0	0		10.1	12.6	12.7	351.3	354.7	324.5	1	325.3	×	
AB	22	m	先天性ぶどう膜欠損	+	0.02	0.15	10	20		10.6	13.7	11.4	307.9	370.1	361.5	1.1	237.2	0	
AD	37	f	糖尿病性網膜症、白内障	+	0.1(0.2)	0.03	0	90		14.4	11.0	16.4	406.5	410.5	408.3	1	338.7	×	
AE	50	f	遠視性乱視	+	0.2	0.1	40	80		10.4	16.7	15.5	291.6	265.2	262.0	0.9	262.6	×	
AG	22	m	白子眼	+	0.15	0.25	60	40		8.3	16.3	8.5	310.4	279.8	358.5	0.7	279.8	0	
АН	21	m	網膜色素変性症	+	0.4(1)	0.7	20	15		11.9	9.6	12.6	376.4	422.9	427.0	0.6	323.2	×	χ μ.
ΑI	18	m	網膜色素変性症		0.1(0.6)	0.5			測定不能	13.8	14.3	20.1	303.9	379.9	326.3	0.5	387.1	0	. 13
ВА	18	m	視神経萎縮	_	0.04	0.04	100	25	左(測定不能)	24.2	28.1	不能	191.9	245.6	不能	1.6	154.2	0	大学大厅
ВС	29	f	網膜色素変性症	+	0.1	0.08	80	80	島状	47.0	41.0	不能	159.8	232.4	不能	1.1	157.2	0	
BD	39	m	網膜色素変性症	+	0.03	(0.05)	28	27		59.0	42.6	不能	117.0	157.9	不能	1.3	71.84	×	. [
BE	45	m	網脈絡膜萎縮		0.1(0.3)	(8.0)	80	85		27.4	27.8	34.2	237.8	247.0	263.3	0.6	279.6	×	
CC	19	m	無水晶体眼	_	(0.15)	(0.15)	45	43		8.4	19.8	16.1	219.7	225.9	221.7	0.9	218	0	
CD	19	m	小眼球、白内障	_	0.3	0.15	10	21		12.5	14.9	16.5	298.7	326.9	272.8	1.1	230	0	•
CE	19	m	虹彩欠損症	_	0.1	0.15	0	0		9.1	15.2	13.6	429.9	516.0	482.0	1.1	335.6	×	-
CF	26	m	網膜色素変性症	+	0.2(0.2)	0.25	85	70		10.7	10.0	8.4	400.8	347.9	397.2	0.5	316.2	. 0	
CG	32	m	黄斑部変性症	+	0.04(0.1)	0.2	5	0		16.8	30.3	42.3	154.3	159.9	148.4	1.1	171.5	0	-
CI	44	m	黄斑部変性症	+	0.06(0.1 5)	(0.2)	0	0		13.9	14.4	18.9	291.8	309.9	310.8	1.1	305.4	0	
CJ	21	m	網膜色素変性症	+	0.06	0.06	88	83		18.2	28.3	29.8	170.0	161.8	103.8	1.2	158.9	0	

3.3 実験の概要

3.4 結果

18名の被験者に実験を実施した。そのうち3名はマウスポインタが見えずにPDF条件の実験ができなかった。残りの15名について、読速度と検索時間について結果を示す。その後、残りの3名について個別

に検討する。なお、被験者のプロフィルは表 4に示す。

3.4.1 読速度について

読速度の結果を表 5に示す。この 結果を分散分析した結果、HTML、 ハードコピー、PDFの平均値間の差 は有意でなかった(図 12)。

CPSによる検討を行った(表 5-2)。 CPSが1.0logMAR以下の群(N=9)と、 1.1logMAR以上の群(N=6)に分けて 検討した。1.0logMAR以下の群で、 HTML、ハードコピー、PDFの3つ のメディア間に有意な差は認められ なかった。1.1logMAR以上群でも、

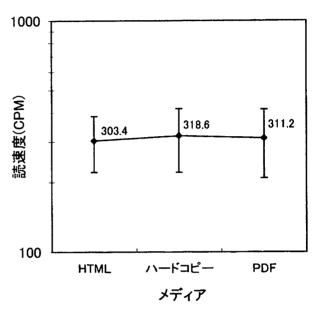


図 12:読速度の比較 (CPM 1分あたりの正読字数)

HTML、ハードコピー、PDFの3つのメディア間に有意な差は認められなかった。

損失率による影響について検討した(表 5-3)。左右の損失率がそれぞれ40以上の群 (N=6)と、それぞれ40未満の群(N=7)に分けて結果を示す。なお、視野を測定できない被験者が1名と、右眼の損失率が90、左眼の損失率が0の被験者が1名いたため、両者を除いた、全体の被験者数は13名になっている。損失率40以上の群でHTML、ハードコピー、PDFの3つのメディア間に有意な差は認められなかった。損失率40未満の群では3つのメディア間の分散分析で有意な差が認められた($\mathbf{F}_{(2,12)}$ = 4.93, p<.05)。さらに、LSD法を用いた多重比較によると、ハードコピーは、HTMLよりも有意に読速度が速かった(MSe=453.9, p<.05)。

PC使用歴の影響について検討した(表 5-4)。PC使用者群では、HTML、ハードコピー、PDFの3つのメディア間で有意な差は認められなかった。PC未使用者群でも、HTML、ハードコピー、PDFの3つのメディア間で統計的に有意な差は認められなかった。

表 5 読速度の結果(単位:CPM)

表 5-1 被験者全体の平均(標準偏差)

	HTM	ハードコピー	PDF
全員(N=15)	303.4(79.8)	318.6(94.8)	311.2(99.4)

表 5-2 CPS別の平均(標準偏差)

	HTM	ハードコピー	PDF
1.0logMAR以下(N=9)	322.0(63.3)	326.0(69.0)	332.1(68.0)
1.1logMAR以上(N=6)	275.4(92.7)	307.4(123.0)	279.9(127.0)

表 5-3 視野損失率別の平均(標準偏差)

	HTM	ハードコピー	PDF
40未満以下(N=7)	315.8(80.1)	351.5(101.1)	332.4(100.1)
40以上以上(N=6)	271.7(73.9)	254.6(56.2)	267.8(94.9)

表 5-4 PC使用別の平均(標準偏差)

	HTM	ハードコピー	PDF
PC使用者(N=9)	273.0(73.4)	284.7(79.5)	277.9(95.1)
PC未使用者(N=6)	348.9(66.2)	369.4(93.2)	361.2(83.6)

3.4.2 検索時間について

次に、検索時間の結果を示す(表 6)。 HTML、ハードコピー、PDFの検索時間を群別に示したものである。

HTML、ハードコピー、PDFの検索時間について(表 $6\cdot1$)、分散分析を行った結果、群間で有意な差が認められた(F(2,28)= 7.40, p<.01)。さらに、LSD法を用いた多重比較によると、HTMLの平均は、ハードコピーの平均とPDFの平均よりも有意に速かった(MSe=15.6, p<.05)。しかし、ハードコピーとPDFの平均値の間には有意な差は認められなかった(図 13)。

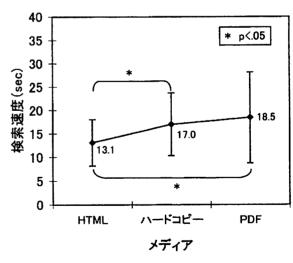


図 13 検索時間の比較

3.4.2.1 CPS別の結果

検索時間をCPSにより1.0logMAR以下の群(N=9)と、1.1logMAR以上の群(N=6)に

表 6 検索時間の結果(単位:sec)

被騒撃令体の重物(両淮恒羊)

表 6・1 彼駅者全体の半均(標準備)	た)		
	HTM	ハードコピー	PDF
全員(N=15)	13.1(4.8)	17.0(6.4)	18.5(9.3))
表 6-2 CPS別の平均(標準偏差)			
	HTM	ハードコピー	PDF
1.0logMAR以下(N=9)	12.8(5.5)	15.4(5.5)	16.1(7.3))
1.1logMAR以上(N=6)	13.5(3.2)	19.5(7.0)	22.1(10.8)
表 6.3 視野損失率別の平均(標準	偏差)		
	HTM	ハードコピー	PDF
40未満以下(N=7)	315.8(80.1)	351.5(101.1)	332.4(100.1)
40以上以上(N=6)	271.7(73.9)	254.6(56.2)	267.8(94.9)
表 6·4 PC使用別の平均(標準偏差	(E)		
	HTM	ハードコピー	PDF
PC使用者(N=9)	12.6(3.3)	18.0(6.5)	19.1(10.3)
PC未使用者(N=6)	13.9(6.3)	15.5(6.0)	17.5(7.6)

分けて比較検討した(表 6.2)。

1.0logMAR以下の群について、HTML、ハードコピー、PDFの3つのメディアにつ いて分散分析を行った結果、3群間で有意な差は認められなかった。

1.1logMAR以上の群の結果について分散分析を行った結果、3群間で有意差が認め られた(F_(2,10)= 5.18, p<.05)。さらに、LSD法を用いた多重比較によると、HTMLは PDFよりも検索時間が有意に短かった(MSe=22.2, p<.05)。

3.4.2.2 損失率別の結果

損失率についても検討を行った。左右の損失率がそれぞれ40以上の群(N=6)と、そ れぞれ40未満の群(N=7)の2群に分けて検討した(表 6·3)。なお、視野を測定できない 被験者が1名と、右眼の損失率が90、左眼の損失率が0の被験者が1名いたため、両者 を除いた、全体の被験者数は13名になっている。

損失率40以上の群で、3つのメディア間で分散分析を行った結果、3群間で有意であ った(F_(2,10)= 4.83, p<.05)。さらに、LSD法を用いた多重比較によると、HTMLはハー ドコピーとPDFのよりも有意に検索時間が短かった(MSe=12.4, p<.05)。

損失率40未満の群では、HTML、ハードコピー、PDFの3群間で有意な差は認めら れなかった。

PC使用歴についても、PC使用者群(N=9)と未使用者群(N=6)の2群に分けて検討した(表 6-4)。

PC使用者群で、HTML、ハードコピー、PDFの3群間で分散分析を行った結果、群間で有意であった($F_{(2,16)}$ = 5.01, p<.05)。さらに、LSD法を用いた多重比較によると、HTMLはPDFよりも有意に検索時間が短かった(MSe=21.9,p<0.5)。また同時にHTMLはハードコピーよりも有意に検索時間が短かった(MSe=21.9,p<0.5)。しかし、ハードコピーとPDFの検索時間の間には有意な差は認められなかった。

PC未使用者群に、HTML、ハードコピー、PDFの3群間で分散分析を行った結果、3群間で有意な差が認められた($F_{(2,10)}$ = 3.53, p<.1)。さらに、LSD法を用いた多重比較によると、HTMLはPDFよりも有意に検索時間が短かった(MSe=5.5836, p<.05)。

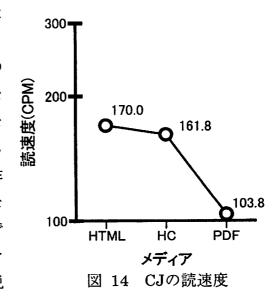
3.5 考察

被験者を各群に分けて分析した結果から次のことが考えられる。

読速度は、分散分析の結果、HTML、ハードコピー、PDFの間で有意な差は認められなかった。文字サイズについてはCPSを補償していたため、有意な差は認められなかったと考えられる。これは、仮説の「CPSを補償していることから、メディア間での相違が無いことも考えられる。」を支持している。このことから、今回の条件において、メディア特性が読速度に与える影響が低いと考えられる。

しかし、PDFは文字サイズが1.1logMAR以上から横スクロールが生じ始める。18

名の内、被験者CJ(CPSは1.18logMAR)個人に注目してみると図 14のようになる。CJは3つのメディアで実験が可能であった中で、CPSの最も大きい被験者である。CJ以上のCPSの被験者はPDF条件での実験ができなかったためデータで比較することはできないが、PDF教材は文字サイズが大きくなるほど横スクロールの幅が大きくなり、操作性が低下する要素が大きいことが明らかなことから、仮にその3名がPDF条件で実験できたとしても、CJと大差ないグラフの形を描くことが予想できる。このことは、仮説



で指摘したとおりであり、文字サイズが大きくなるにつれてPDFの操作性が低下することが示唆される。

損失率40未満の群で、ハードコピーの読速度がHTMLよりも速かった。図 15は損失率40未満群の読速度のグラフである。グラフ中で大きな菱形(◇)の平均値は、山型を示している。この損失率40未満群7名の内、5名が平均値と同様の緩やかな山型を示している。5名の内、CPSが1.1logMARの者が4名も含まれている。

そこで、CPSが1.1logMAR の群の読速度について検討し てみた。それが図 16である。 その結果、CPS1.1logMAR群5 名の中で山形を示さなかった のはわずか1名しかいなかった。 このことがこの結果に影響を 与えていると考えられる。つ まり、CPSが1.1logMARの HTML教材の条件は、1行文字 数が16文字で表示されており、 一方、ハードコピーとPDFは1 行14文字で表されていて、3つ のメディアの表示条件がほぼ 等しい環境になっているので ある。このことから 1.1logMARの条件は読書する 上で3つのメディアの特性が現

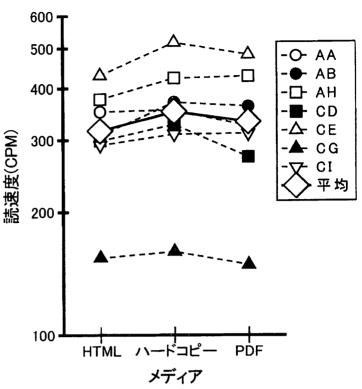


図 15 損失率40未満群の読速度

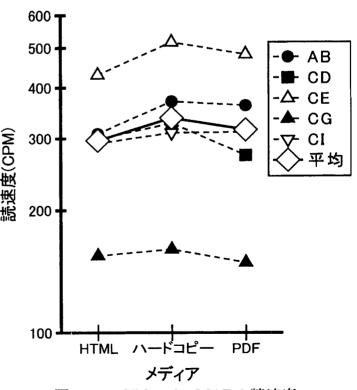


図 16 CPS1.1logMARの読速度

れやすい環境だと考えられるのである。そのため、CPSが1.1logMARの群では、最も 読み慣れているハードコピー教材の読速度が他の電子メディアよりも速くなっている のではないかと考えられる。損失率40未満の群でハードコピーが電子メディアより読 速度が速くなった原因については、以上のように推察できる。しかし、1.1logMARと いうのは絶対的な条件だとは言えないだろう。本実験では文字サイズの設定が、偶然 1.1logMARの条件でハードコピーと電子メディアで同じようなレイアウト条件になっ たに過ぎず、常に1.1logMARの条件でハードコピーが電子メディアと同じレイアウト 条件になり、読速度が速くなるとは言えないからである。

検索時間の全体の傾向は、HTMLがハードコピーやPDFよりも5%の有意差で速いことが分かった。内省報告では、9名(60%)の者がHTML教材のアンカー文字が便利だったと答えたことから、HTML文書表示ツールを利用してアンカー文字が反転する効果を与えたことが、この結果に影響していることが考えられる。このことは、同じハイパリンクを用いているにも関わらずHTMLがPDFよりも速く検索できることからも裏付けられる。この他にHTMLがPDFよりも優れた結果を出した理由として、HTMLの目次は1つのファイル(1ページ)からなっているのに対し、PDFの目次はその特性上2ページからなっていること、マウスボインタは黒の最大に設定していたにもかかわらず、PDF文書表示上ではPDF特有の自の指型マウスポインタになってしまい視認性が低下したことなどが考えられる。これらのことから、HTML教材の有意性について個々の要因についてまで言及できないが、表示ツールを利用したことが大きな要因の1つとして関わっていることが考えられる。

次にCPSの影響について分析する。CPSは1.0と1.1logMARの間を境にした。これはちょうどPDF画面で横スクロールが始まる境目である。1.0logMAR以下の群では特に有意な差は認められなかった。このことは、比較的CPSが小さい場合(1.0logMAR以下)はメディアの違いが検索のパフォーマンスに影響を与えにくいことを示しているものであろう。一方、1.1logMAR以上の群においては、HTMLがPDFよりも速くなるという結果であった。前述のHTMLとPDFのメディアの特性と考え合わせると、1.1logMAR以上の群ではアンカー文字反転機能やマウスの色の恩恵を受けやすいことが考えられる。

次に視野の損失率により2群に分けて分析した。損失率など視野にもとづいた分け 方については、先行研究が見当たらなかったので、左右のそれぞれの損失率により、 便宜上40を境界として2群に分けた。損失率が40以上の群では、HTMLがPDFよりも 速かった。損失率が高いと画面を広角でとらえることができない。そのため、アンカ 一文字の反転やマウスポインタの色などの要因が有効に働き、PDFよりもマウスポイ ンタを有効に活用できる結果だと考えられる。

損失率が40未満の群ではメディア間で有意差はなかった。これは画面を広角でとらえることができるため、常に画面の大部分とマウスポインタが視野に入っており、HTMLのアンカー文字の反転やマウスの色が検索のパフォーマンスに影響を与えないことが推測できる。この結果をより明確化するためには、その時にマウスポインタをどのような眼球の追従運動により追跡しているか、限られた視野の中でアンカー文字やマウスポインタなどをどのように発見しているかなどの基礎研究が必要である。

次にPCの使用歴から、使用者群と未使用者群の2群に分けて検討した。使用者群ではHTMLがハードコピーとPDFよりも検索時間が速かった。当初の予測では、検索作業は読書作業に比べ、マウスの移動やクリックなどのマウス操作の要因が大きく影響するため、使用者群はハードコピーよりも電子化教材の方が速くなることが予想された。しかし、結果はHTMLが最も速くハードコピーとPDFはあまり変わらなかった。このことから、PC使用者についても前述したHTMLとPDFの操作性の違いが影響を及ばしているものと予想される。PC未使用群では、HTMLがPDFよりも速いという結果であった。普段から使い慣れたハードコピーと、ほとんど使用経験の無いHTMLの検索作業はほぼ同じであった。普段からPCを使用しておらず、操作に熟練していない未使用者群でHTMLの検索時間がハードコピーと変わりないことは、HTML教材の操作性は高く評価でき、スムーズに導入できることが予想できる。以上のような結果から、PDF教材とHTML教材の導入を検討する場合、検索時間から見る限り、HTMLはPDFよりも操作性がよく、PC未使用者でもハードコピー教材と同様な検索時間を示していることから、導入には問題が少ないと考えられる。

3.6 事例による検討

ここでは、PDF条件での実験ができなかったBA、BC、BDの3名について取り上げる。3名とも、HTMLとハードコピーの条件で実験は可能であったが、PDFでの実験ができなかった。その理由は、PDF条件ではマウスポインタが見えなかったためである。つまり、前述したように、マウスポインタの形状や配色などを

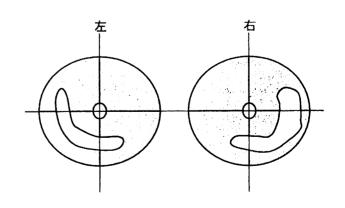


図 17 BCの視野の模式図 (白い部分が見えている)

設定していても、PDFを見るときには白色の指状のマウスポインタになってしまい、 今回のような白い背景の条件ではマウスポインタを見つけにくくなってしまうのであ る。

被験者BCはCPSが1.1logMARと比較的良い方であるが、マウスを同定できなかった。BCは視野の中心と耳側下方が島状に残っている(図 17)。中心の視野はわずかに残っているものの、中心部と周辺部の間に大きく視野の欠落した部分があり、マウスポインタが同定できなかったものと考えられる。BAとBDはCPSがそれぞれ1.58logMAR、1.28logMARとかなり大きいため、サイズに条件のあるマウスポインタは、見つけにくかったものと考えられる。

読速度は、3名ともHTMLよりもハードコピーの方が速かった。これは、3名とも文字が大きい条件であり、HTMLで表示した場合、1画面に表示される1行文字数が減るり、回り込みの量が増えてしまい、結果的に行数が増えてしまい、縦のスクロールの操作が多くなるためではないかと予想される。

検索時間は、1年程度PCを利用して操作になれている被験者BAのみ、HTMLがハードコピーよりも速くなった。BCはPC利用者であるが、今年度になって、授業で週に1時間しか程度っていないため、操作になれていないことが考えられる。BDはPCを扱ったことがないためであると考えられる。

この3名は、PDF教材は視力や視野の 状態から利用できなかった。しかし、表 示ツールを用いたHTMLは、アンカー文 字を反転させることにより、マウスポイ ンタが見えにくくても、十分に操作する ことができた(図 18)。このような工夫は、 電子化教材の導入を考えていく上で、特 に最近のGUIやマウスの操作が問題視さ れる中、それらの障壁を取り払う1つの 方法を示していると考えられる。

- 1. はじめに
- 2. 教材集
- 3. 研究報告書
- 1. はじめに
- 2. 教材集
- 3. 研究報告書
 - 図 18 アンカー文字の表示の違い 上段はホームページで多く利用さ れている方法。

下段はアンカー文字を反転させる方法。

3.7 まとめ

HTML、ハードコピー、PDFで読速度と検索時間を測定し比較した結果以下のことが示唆された。

- 1. 読速度は、HTML、ハードコピー、PDFのメディア間で有意な差は認められなかった。電子化教材は条件が整えばハードコピー教材と変わらない読速度が保たれること、CPSを補償することによりメディアの違いが読速度に及ぼす影響は小さいことが示唆された。
- 2. 検索時間は、HTMLはハードコピーとPDFより速いことが分かった。その要因の1つとしてHTML文書表示ツールを用いたことが考えられた。
- 3. 検索時間は、CPSが1.0logMAR以下の群ではメディア間に有意差は認められなかったが、1.1logMAR以上の群では、HTMLがPDFよりも速かった。このことから、CPSが大きい群では、アンカー文字の反転やマウスポインタの色が検索のパフォーマンスに及ぼす影響が大きいことが示唆された。
- 4. 左右それぞれの損失率が40以上の群では、検索時間は、HTMLがPDFよりも速かった。左右それぞれの損失率が40未満の群ではメディア間で差は認められなかった。このことから、視野の障害が大きい群では、アンカー文字の反転やマウスポインタの色が検索のパフォーマンスに及ぼす影響が大きいことが示唆された。
- 5. PC使用者群では、HTMLはハードコピーとPDFよりも検索時間が短かったが、ハードコピーとPDFの間には差は認められなかった。PC未使用者では、HTMLがPDFよりも速かった。HTMLの操作性の良さはPC未使用者でも現れることが示唆された。
- 6. CPSが約1.2logMAR以上、あるいは視野に問題があるなど、視機能によっては PDFのマウスポインタを見つけることができない場合があることが分かった。 しかし、HTMLにおいてアンカー文字を反転させるなどの工夫をすることで、 電子化教材を利用することが可能になることが分かった。
- 7. CJが描いたグラフから、文字サイズを大きくした場合、PDFでは読速度が著し く低下することが示唆された。

これらのことから、盲学校への電子化教材の導入を考えた場合、HTML教材が十分に活用できることが示唆された。さらに今回使用したような表示ツールを用いたり、CSSを用いたりすることでロービジョンにとって、さらに望ましい読書環境が作り出せることが分かった。

4 HTMLの展望

4.1 HTMLの現状

HTMLは、ドキュメント(ハイパテキスト)の構造を記述し、情報を関連づける(ハイパリンク)ためのマークアップ言語であり、WWWの普及にともない発展してきた。

HTMLの公式規格には、HTML2.0、HTML3.2、HTML4.0の3つがある。

HTML2.0は、1995年11月に、インターネットで利用される技術を標準化する組織であるIETF(Internet Engineering Task Force)が共通化しようとして策定したものであり、現在、ほとんどのWebブラウザやHTML文書を扱うアプリケーションがサポートしている。この規格に沿って作成されたHTML文書の互換性は極めて高いものがある反面、策定時期が古く、多くの機能不足が指摘されている。

HTML3.2は、WWWの標準化推進のため設けられた国際的な学術研究組織であるW3C(World Wide Web Consortium)において、1997年1月に勧告された仕様である。Webブラウザのメーカーによって開発・普及していた色やフォント、表に関わる部分を拡張した規格であり、HTML2.0に次いで互換性も高い。なお、HTML3.2はHTML3.0に上書きされて現在に至っている。したがって、HTML3.0という規格は現在存在しない。

HTML4.0は、HTML3.2を大幅に拡張した規格であり、1997年12月にW3Cより勧告された。フレームや本研究で使用したスクリプト言語(JavaScriptなど)の正式サポート、およびマルチメディアデータの扱いが改善されているとともに、HTML文書の見栄えに関する要素を分離し、HTMLは本来の文書構造記述に専念できるようになっている。しかし、この規格を完全にサポートするWebブラウザやアプリケーションは現在の所ほとんどない。また、ブラウザソフトメーカーで独自に開発された拡張機能が増え、各ブラウザで共通して利用できるHTMLコンテンツ制作の障壁になっている。

4.2 アクセシビリティへの取り組み

このように情報化社会を支える仕組みや規格は順調に発達している。これらの情報 通信技術はアクセシビリティについても国際的な規定ができており(References on Web Accessibility)、国内においてもそれは自由に入手できる(すべての人にアクセス 可能なHTML文書を書く)。

「Webページのアクセシビリティ」とは、「利用者がどのような装置を使っていても、どのような環境下にあっても、どのような障害を持っていても、Webページ上の情報に同じようにアクセスできて、利用できることをいう。現在のような高度情報化社会においては、電子情報、電子メディア、コンピュータへのアクセスは、所得、障害、居住場所にかかわらず、等しく保証されなければならない」(References on Web Accessibility)としている。

このような、国際的な動きに同調して、国としてもアクセシビリティに対する施策が行われている(総理府, 1998)。障害のある人にとっても使いやすいコンピュータや処理機器の開発普及への取り組みとして、政府は、平成元年に「障害者等対応情報機器開発普及推進委員会」を設置し、平成2年6月に「情報処理機器アクセシビリティ指針」を公表した。さらに平成7年4月にはその後の情報処理機器の技術の進展に合わせて改訂を行い、「障害者等情報処理機器アクセシビリティ指針」として告示している。平成9年5月16日に閣議決定された「経済構造の変革と創造のための行動計画」では、医療・福祉関連分野と情報通信関連分野が新規・成長15分野の1つとして位置付けられ、その中で「情報バリアフリー」な環境整備に配慮すること、ニーズの多様化や高齢化の進展などの医療福祉をめぐる状況の変化に対応する上で情報化が有効な手段となりうることが指摘されている。こうした観点から、具体的な施策として「障害者や高齢者が扱いやすい情報通信機器の開発・普及を計る」「情報通信の利用に関するバリアフリー化を推進するため、情報アクセスに関するガイドラインの策定を推進する」「高齢者・障害者のための機能代行・支援通信システム技術の研究開発を推進する」等の行動計画が策定されている。

平成9年11月から平成10年6月まで開催されたライフサポート情報通信システム推進研究会が策定した電気通信設備のアクセシビリティ指針から視覚障害に関する項目を挙げると、

- 1. 障害に関わらず入力を可能とするための基準
- (1)視覚に頼らなくても入力可能であること。
- (2)色の識別を必要とせずに入力可能であること。
- 2, 障害に関わらず出力結果を利用するための基準
- (1)視覚的な情報を視覚に頼らない機能で提示可能であること。
- (2)動画方式の情報を静止させる機能を提供可能であること。

などである。このように情報弱者に対するアクセシビリティについての取り組みがされている。

しかし、視覚障害に対するこれらの取り組みの多くの部分は、全盲のユーザーに対するものである。平成8年の調査によると、視覚障害者のうち全盲と思われる身体障害者手帳1級の所持者が視覚障害者に占める割合は27.1%であり、その他の72.9%はロービジョンであることが予想できる。これだけ視覚障害者にしめるロービジョンの割合が多いにも関わらず、これらの取り組みの中で、ロービジョンのアクセシビリティについては、問題意識が薄いといわざるを得ない。

4.3 HTMLの今後

W3Cで行われている新しい規格化の中でも、XMLとRDF(Resource Description Framework)は、特に重要になっている。XMLはSGMLとHTMLから発展した新しいWWW文書記述言語であり、今後のWWWアプリケーションシステムを構築する上で中心的な規格となることを指摘している(保田、1999)。これに対し、HTMLは、WWWブラウザ上での文書の表現に重きをおいた言語であり、その簡単さから広く普及したが、WWWを核とする技術が発展し、応用サービスも多様化してきた現在、HTMLで記述できる情報には限界が生じてきた。HTMLは、文書を閲覧するための言語で、例えば電子商取引のためのデータや、基幹業務システムで扱うデータをWWWサーバを介して交換するシステムでは、データの属性など各種の業務で扱う情報までを記述することが困難になってきている。

ここで登場してきたのがXMLである。ある意味でXMLはHTMLの後継となるWWWのページ記述言語のように思われるむきもあるが、それはXMLの一側面に過ぎない。XMLは、WWW上に存在する情報を記述するための汎用的なマークアップ言語であり、SGMLと同様に文書の構造に基づき記述する言語である。XMLは、WWW上のコンピュータシステムを結合し、文書の交換のみならず、各種業務システムの統合を実現するための新規格として注目されている(加藤ら, 1999)。

現在、WWWページの記述に使われているHTMLは、ブラウザで文書を表示することを目的とした言語であるため、レイアウトを中心とした記述機能を有する。HTML文書の最終目的は人がブラウザで内容を確認することであり、HTML文書の内容をコンピュータで理解しようとすると、レイアウト解析などの技術が必要となり、大変な困難を伴う。このため、コンピュータ間で文書やデータを交換する形式として不向きであった。

XMLでは、文書やデータの構造に着目し、中に含まれる情報に意味を付加して記述するため、コンピュータでも容易に内容が処理できる形式となっている。そのため、

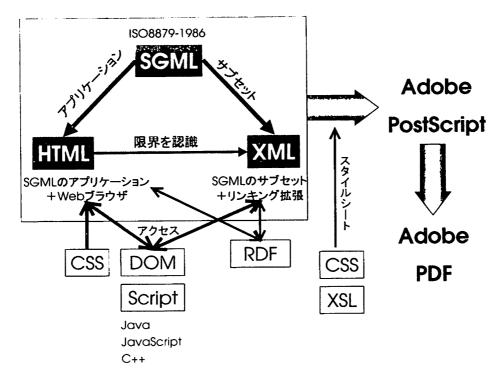


図 19 XML関連の各標準の関係(東, 1999)

WWW上のコンテンツを取得し、データベース化したり、業務システムへ情報を流したりすることが容易に実現できる。すなわち、XML文書は、ブラウザで人が見るだけでなく、バックエンドの業務システムにまで情報を伝えていくことができる。

このため、WWW上のコンピュータシステムを結合し、文書交換のみならず、各種業務システムをシームレスに接続していくための新規格として将来有望視されている。 そのほかにも、XMLはデータを取得して動作を決定するデータ駆動型のクライアントアプリケーション用のデータ記述言語として、また様々なコンテンツの管理や再

利用を可能とするための記述言語として検討がなされている(加藤ら, 1999)。

次に、今回の実験で使用したHTMLやPDFとXMLの関係と今後について概観してみる(図 19)。東(1999)は、PDFのWeb利用においての優位性はペーパーメディアとの連携であり、XMLはWebが文書の単なる配送手段から業務システムをネットワークで実現するためのインフラストラクチャへと変わることを指摘している。その中で、HTMLのメリットは、タグを記述すればすぐに表示できることであり、HTMLとXMLの今後の関係で考えられることの1つに、HTMLのメリットである固定タグセットを使用した表示やプリントという機能はそのまま残し、固定タグセットにはない拡張機能部分について、XMLで記述したデータをHTML文書の中に埋め込むという利用方法があると指摘している。

したがって、今後、XMLが完全にHTMLに取って代わるというものではなく、 XMLのなかでHTMLのメリットが生き続けると考えられる。また、東(1999)はXML でもHTML同様のスクリプト言語をサポートするとしている。したがって、今回作成 したような表示ツールをスクリプト言語により開発することがXML中でも可能であ る。

4.4 HTML教材普及における課題

HTML教材は、ハードコピー教材と同等以上のパフォーマンス、HTMLの現状や今後の展望などと考え合わせると、リーディングエイド、あるいは教材提供手段として有望な方法の1つであることが分かってきた。しかしながら、HTML教材を普及させるには今後乗り越えなければならない課題も残っている。ここでは、今後改善を要する課題について取り上げる。

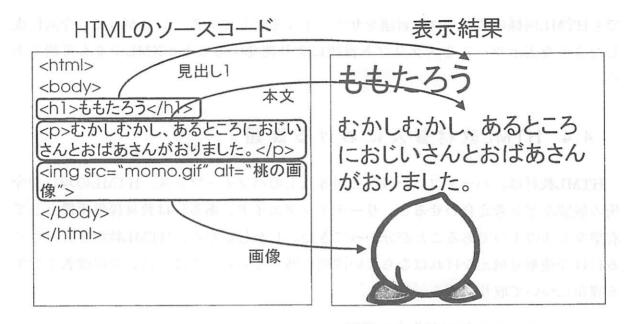
4.4.1 HTML教材制作者の問題

HTMLを用いて、文書を作ろうとすると、(1)ワープロソフトを用いて変換する方法、(2)ホームページ作成ソフトを使用する方法、(3)エディタを用いてコードを記述する方法の3つの方法がある(岡崎, 1999)。一見(1)(2)の方法が簡単そうであるが、これらの方法で作られたHTML文書は、出力の際にHTML形式への変換を行うだけなので、HTMLの機能を十分に使いこなすのは用意とは言えない部分も出てくる(岡崎, 1999)。したがって、本来のHTMLの規格から判断する限り最善とは言い難い。

そこで、ここではHTML教材の制作者の問題に入るために、HTMLやCSSの原理を 簡単に触れておくことにする。

4.4.1.1 HTML文書の構造

HTMLは文書を構造化する言語である。1つの文書があった場合に、「ここが見出し1」「ここがリスト」という具合に、文書中の要素を定義づけるものである。そして、その定義付けに用いる印を夕グ(tag)と読んでいる。図 20は、HTMLを用いて文書を構造化している例を示している。まず、<html> </html>という夕グによりHTML文書の最外層をマークアップし「この文書はHTML文書であること」を定義している。そして、<body> </body>タグに挟まれた部分が本文である。ここで分かるように、見出し1は、<h1> </h1>、本文は、 などのように夕グが決まっている。ここがHTMLの限界である。XMLでは、この夕グを作り手の都合の良いように設定することができるのである。しかし、簡単な、読み物や教材を作る程度であればHTMLで十分であり、この辺りの使い分けにより、今後XMLとHTMLは共存していくと言われているのである。



で、囲まれた部分が、文書を構造化している。

HTMLには、文字サイズ、行間隔、配色などのレイアウトに関する情報は、基本的に含まれない。

図 20 HTMLによる文書の構造化の例

図中で、「HTMLには、文字サイズ、行間隔、配色などのレイアウトに関する情報は、基本的に含まれない。」と説明している。これは、HTMLにはレイアウトの機能は与えられていないにも関わらず、タグの拡張機能や、ブラウザソフトの機能である程度のレイアウトを制御することができるため、「基本的に」と表現してある。

それでは、HTMLで記された文書のレイアウトを制御しようとすると、どうすれば良いのだろうか。それを制御するのが、CSS(Cascade Style Sheet)である。図 21はHTMLとCSSの関係を示している。図中ではタグの説明がしてある。このようにCSSは、タグのレイアウトを制御する。HTMLによる構造化部分とCSSによるレイアウト設定部分を分けて記述することは、管理を容易にし、コードの理解を容易にする。また、タグ中にレイアウト情報を入れてしまうと、ロービジョンの読みやすいレイアウト情報を書き込んだCSSファイルをあてても、それが反映しなくなる恐れがある。このような理由からもレイアウト情報はCSSに任せるべきである。

このように規則にのっとり、整然としたHTMLやCSSを記述しようとすると、エディタなどでソースコードを書くのが最も望ましい方法なのである。



図 21 HTMLとCSSの関係

4.4.1.2 HTML教材制作者の養成

上記のようにエディタでHTML教材を記述しようとすると、多少のHTMLやCSSの知識が必要になる。しかし、実際に教材や読み物をHTMLで記述するために必要なタグの種類は十数種類である。今回制作した教材のサンプルで

body> </body>中で用いられているタグは、<h1> </h1> 、<h2> </h2> 、<h3> </h3> 、 、<a> 、 、 、 、

たびって、これらのタグの使用方法を理解すれば、HTML文書を書くことは可能なのである。岡崎(1999)は、教師が自分の必要に応じて、容易にドキュメントが作成できるように、HTMLドキュメント作成のガイドラインなどの整備を進める必要性を指摘しており、教師が必要に応じてHTML教材を作ることが一般的に求められつつあることを示している。これがロービジョンの教材となると、本研究の結果からも、その重要性は一段と増してくると考えられる。

さらに、ロービジョンの読書環境に適した、見やすいレイアウトの設定をしたい場合は今回試作した表示ツールを用いるか、CSSを覚える必要がある。しかし、こちらは、制作者全員が覚える必要があるわけではない。図 22で示すように、1つのCSSファイルを作っておいて、それをそれぞれのHTMLファイルから参照するようにすれ

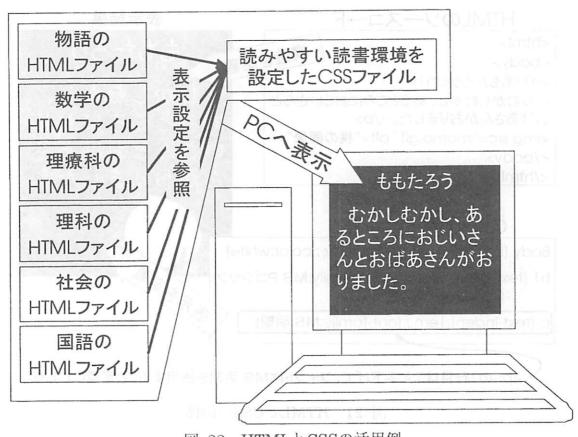


図 22 HTMLとCSSの活用例

ばいいのである。今回の実験でもHTML教材の有効性が示唆されており、教材の電子 化を考えた場合、十数種類のタグを覚えてHTML教材の製作法を学ぶことが望ましい。 もちろん、CSSファイルの書き方を覚えておくと、ロービジョン一人一人の読書環境 に適したレイアウトを作り出すことができるので、それも大切である。

4.4.2 VDT作業の問題

PC作業を中心とするVDT作業を行う上で、それが身体へ及ぼす影響について正しく理解し、回避する方法を知っておくことは大切なことである。特にロービジョンのVDT作業による影響、特に眼への影響は、晴眼者以上に危惧される。岡田(1990)はロービジョンのVDT作業の影響について自覚症状の聞き取り調査をした結果、VDT作業による負荷が晴眼者に比べロービジョンでは大きくなる傾向にあることを報告している。VDT作業する上では、電磁波の問題、ドライアイの問題、不定愁訴の問題などが挙げられる。

4.4.2.1 電磁波の問題

電磁波は、PCに限らず世の中の家電からは少なからず生じている。電磁波に関する健康被害はいくつか報告されている。また電磁波と病気の発生率との相関関係を指

摘した疫学調査もされている。しかし電磁波と病気の間の因果関係についてははっきりしたことは分かっていない。WHOは1996年5月電磁波の解明のために330万ドルの予算を付けて、5年計画で大規模な研究プロジェクトをスタートさせている(神立,1997)。研究は進んでいるが、その結果が出るまでにもPCの利用は欠かせない。そこで現在分かっている有効な対策を挙げる。

- 1. なるべく離れる;電磁波を浴びる量は、電磁波の発生源からの距離の2乗に反 比例する。したがって、なるべくディスプレイから離れて作業することが望ま れる。したがって、ロービジョンにとっては、ディスプレイから離れても作業 ができるように文字を拡大したり、配色を調節したりすることが重要になる。
- 2. できるだけ長時間使わない; VDT作業は連続して行わず、休憩を入れることが 望ましい。CFF(Critical Fusion Frequency)値を用いた眼精疲労の実験による と、5人のロービジョンに50分間の一位数の連続加算作業を行わせ、10分毎に CFF値を測定した結果、およそ30分後に休憩をとることが目安であることが報 告もされている(高橋ら, 1990)。労働省の基準では1時間に15分程度の休憩が求 められている。また、1日の操作業時間にも配慮すべきである。
- 3. 電磁波自主規制;(社)日本電子工業振興協会は、1994年にブラウン管のPCの電磁波に関する自主基準「情報処理機器用表示装置の低周波電磁界に関するガイドライン」を制定し、1996年8月に改訂した。これには1998年1月以降製造される機器は全面適用されることになっている。
- 4. PCの置き場所;目の前のCRT対策を講じると同時に、横や後ろからの電磁波にも注意しなければ意味がない。したがって、PCはデスクの下に置いたり、背中合わせにデスクを配置したりするなどしてOA機器全体の設置にも注意する必要がある。
- 5. 液晶ディスプレイの使用;消費電力がCRTよりも少ない液晶ディスプレイを使用するとそれだけで電磁波は抑制できる。液晶ディスプレイを十分に利用することができるのであればそちらを進めるべきである。

4.4.2.2 ドライアイの問題

「VDT作業を行うと目が疲れる」という現象を体験したことのある人は多いだろう。 その原因の1つにドライアイが考えられている(神立, 1997)。以下にドライアイに対 する対策についてまとめる。

1. まばたきの現象;そもそもVDT作業を行うだけでまばたきの回数が4分の1に減少することが分かっている。すなわちVDT作業を行っている最中は眼が乾燥し

ている状態にある。これは見やすい環境を作ることで改善されるといわれている。CRT画面には直射日光が当たらないようにする、照明はグレアを生じないように蛍光灯フィルターなどを用いて適切に調節する、文字サイズや配色などのレイアウトを適切に設定する、などが対策として挙げられる。したがって、今回試作したツールやCSSなどを用いてロービジョンが読みやすいレイアウトを構築するだけでこの問題をある程度改善できる可能性がある。

- 2. 画面を見上げる姿勢; PCを設置するときに本体の上にCRTを置くケースをよく見かける。このような上転眼位での注視作業は、眼表面積の拡大を招き(外山, 1992)、涙液蒸発量増加によるドライアイの発症を促す要因ともなり(坪田, 1992)、眼科的に好ましくない。これらのことから作業姿勢は下転眼位のもとに画面注視が行われることが好ましいことが分かっている(鈴木, 1993)。したがって、できるだけディスプレイは上から見下ろす位置に設置することが望ましい。
- 3. 部屋の乾燥;特に冬季エアコンにより暖をとるケースは気を付ける必要がある。 例えば室温10℃で湿度50%の時に、暖房を入れて室温を20℃まで上げると湿度 は25%まで下がるという報告がある(神立, 1997)。空気が乾燥すると当然眼も 乾燥するので加湿器などで60%以上85%以下の範囲で湿度を調節することが望 まれる(神立, 1997)。

4.4.2.3 不定愁訴の問題

VDT作業を行うときは大きな身体運動を伴わないため同じ姿勢を維持し続けてしまう。このことから特に肩こりや頭痛、腰痛などの愁訴を訴えることがある。これは、同じ姿勢を長時間続けることによる筋の疲労などが主な原因と考えられる。したがって、適度な間隔で席を立ち体をまんべんなく動かしたり、ストレッチをしたりするなどして筋の血流を改善することが大切である(神立, 1997)。また、あん摩やマッサージは局所の血流を改善する効果が認められているので、筋性の肩こりや頭痛には効果が期待できる。

以上のようにVDT作業による体調不良は多くの要素により構成されている。ここでは触れなかったが、読書や学習によるストレスが症状を生起させることも考えられる。したがって、ロービジョンの意見を十分に考慮し対応する必要がある。HTML教材は今回試作した表示ツールやCSSと合わせて使用するとかなり自由にレイアウトを調節でき、VDT作業による疲労を軽減するための有効な提示方法の1つとして十分に効果的であることがうかがえる。

ロービジョンは複数の眼疾患を併せ持っている場合が多く、見え方も多彩である (大川原ら, 1999)。したがって、普段から視力や視野などの変化に注意を払い、急激 な視力の低下や視野の変化、飛蚊症などに対して適切な対応が望まれる。

4.5 まとめ

HTMLの現状と展望、HTML教材普及の課題について検討した。このことから次のことが分かった。

- 1. 情報化社会において、情報弱者への補償が国やW3Cなどの機関で検討されている。しかし、視覚障害者に着目すると、ロービジョンに対する検討内容が少ないことが分かった。
- 2. HTMLは、今後XMLに完全に移行するというより、XMLの中で残っていくことが分かった。またXMLの中でもJavaScriptのようなスクリプト言語が使用できることが確認できた。したがって今後も、HTMLやXMLを利用した教材の作成が可能で、さらに今回試作したような機能を持った表示ツールを使用することも可能であることが確認できた。
- 3. HTML教材の普及に関しては、教師がHTMLやCSSの特徴を理解することが望ましいと思われた。HTML教材については十数種類のタグを覚えることで作成できる。またレイアウトは今回試作した表示ツールを使用したり、独自にCSSファイルを作成したりすることで手軽に設定できることが分かった。したがって教師がHTML教材を作成することは想像以上に容易であると考えられた。
- 4. VDT作業に関しては、特に電磁波、ドライアイ、不定愁訴などに注意し、30分に一度の休憩が目安であることが分かった。またロービジョンのVDT作業については特に注意が必要であることがうかがえた。

HTMLの規格は今後もXMLと共存していくことが確認できた。したがって、今後もHTML教材を普及させていくことは世の趨勢やHTMLの利便性から考えると適当であると考えられる。また、利用者のリスク管理や作り手の育成などの条件が整うのであれば、HTML教材の普及は比較的容易であると考えられる。

5 おわりに

新学習指導要領では、小学校の総合学習から、高等学校の情報へと一貫して情報リテラシ教育、コンピュータ教育が盛り込まれている。これらのことから考えても、本研究のような電子メディアを用いた教材の研究は大きな潮流となっていくことが予想される。また、多様化する障害の中で、一人一人に応じた読書環境を手軽に構築する事を可能にした表示ツールは画期的であるといえる。更に、インターネット上でこれらの教材を提供できることを考えれば、訪問教育や在宅教育、ロービジョンの生涯学習への発展も期待できる。

本研究では、ロービジョンの読書パフォーマンスを阻害する要因をコントロールして、ロービジョンにとって理想的なレイアウト設定でHTML教材を表示するツールをJavaScriptで試作した。その結果、ロービジョンの読書パフォーマンスを左右する要因をコントロールして表示できるツールが完成した。またカーソルキーのみでの環境設定と、アンカー文字の反転機能を備えたアクセシビリティの高いツールになった。本表示ツールは、ハードコピー教材の作成にも利用できることが分かった。また、いくつかの教材を作成し表示ツールとHTML教材の可能性を示した。

本表示ツールとHTML教材の有効性を確かめるための簡単な実験を行った。その結果、表示ツールを用いたHTMLはハードコピーと同等の読書パフォーマンスを保つことが示唆された。また、検索作業については、CPSが大きい群や視野の障害が大きい群でハードコピーよりもHTMLの方がパフォーマンスが優れており、その要因の1つとして表示ツールを利用した効果が示唆された。今後の情報教育の中でWebコンテンツの操作方法に慣れた生徒が増えてくると、HTML教材の重要性は増してくることがうかがえる。

また、SGML系のマークアップ言語の流れを見ても、XMLがすぐに全盛になるのではなくHTMLと共存していくことが予想でき、スクリプト言語に対応していくことも確認できた。したがって、今回のような表示ツールとマークアップ言語を活用した電子化教材の考え方は将来にわたって利用できることが確認できた。

拡大教材、拡大教科書の歴史は古く、手書きの時代を経て平成4年に拡大教科書が 出版されるまでに至っている。しかし、一人一人の見え方が多様であるロービジョン に対応するためにはプライベートサービスにより、それぞれの見え方に応じた拡大教 材(拡大教科書)を作成していかなければならないことが指摘されている(大川原ら, 1999)。 本研究でHTML教材はPCでの利用のみでなく、ハードコピー教材を作成する方法としても優れていることが分かり、プライベートサービスをカバーする方法として有効であることがうかがえる。したがって、一人一人の見え方が多様なロービジョンの拡大教材(拡大教科書)づくりの1つの方法としてロービジョン教育の世界で活用されることが望まれる。教科書の著作権などの問題もあるが、それぞれのロービジョンの見え方に応じた教材作成を容易にする電子化教材の普及と発展に今後も努めていきたい。

6 謝辞

今年度、研究所で研修できたことは私にとって大変意義があった。また、弱視教育の世界に対しても、1つの教材の表示方法を示せたことは意義あることであると自負している。今後も、弱視教育の視点から学校教育・社会教育・生涯学習に対して前向きに取り組んでいきたい。

本研修にあたり協力いただいたみなさん

- ◆ 貴重な時間を割いて指導していただいた大内先生、千田部長はじめ視覚障害教育 研究部の先生方、弱視教育に対しアドバイスいただいた新井先生
- ❖ 視覚科学について最新の情報を提供していただいき、表示ツールについてご助言いただいた東京女子大学の小田先生はじめ小田研究室のみなさん
- ◆ 研修の場を与えてくださった愛媛県教育委員会
- ◆ 研修を地元より支えてくださった橘校長はじめ愛媛県立松山盲学校の職員のみなさん。特に実験の受け入れや資料収集に応じてくださった先生方
- ◆ 技術的な相談に乗ってくださった、愛媛大学の村田先生とBu.systemのみなさん
- ◆ なによりも地元から暖かく見つめてくれた氏間由美さん以上の方々に心より感謝申し上げます。

連絡先

国立特殊教育総合研究所

〒239-0841 神奈川県横須賀市野比5-1-1

TEL: 0468 48 4121

FAX: 0468 49 5563

URL: http://www.nise.go.jp

愛媛県立松山盲学校

〒791-8016 愛媛県松山市久万ノ台112

TEL: 089-922-3655

FAX: 089-922-2893

氏間和仁

E-mail: ujiman@dokidoki.ne.jp

この報告書に関する詳細な情報は下記のURLでご覧いただけます。 http://www.busystem.palclub.ne.jp/usr/ujiman/

7 用語集

1. CGI (Common Gateway Interface)

WWWサーバのシステムにおいて、HTMLから外部のプログラムを起動して、その 結果をWWWクライアントへ返すための仕組み。HTMLだけでは記述できないような 動的なページを作成することができる。

たとえば、ユーザーの入力したフォームのデータをサーバ側へ読み込んだり、ユーザーからの指令にしたがって表示内容やその応答を変えたりするために使用される。 また、ユーザーのアクセスカウンタ(そのページがアクセスされた回数を表わす数値のカウンタ)なども、このCGIを使って実現されている。

WWWサーバと、CGIを通して起動されるプログラムは、通常は標準入出力を使ってデータをやり取りする。(http://www.ascii.co.jp/pb/ghelp/14/001421.html)

2. cookie(クッキー)

JavaScriptやCGIなどが、クライアントコンピュータのローカルファイルへアクセスする方式。

3. CSS (Cascade Style Sheet)

Webページにレイアウト機能を追加するために考案されたフォーマット規格。CSSと略記されることもある。現在、HTML3.2のドラフト案として、W3Cで仕様が策定されている。

Webページを記述するためのHTMLは、もともとは文書の論理構造を記述する SGMLがベースになっており、文書の表示機能はあまり積極的には規定していなかった。しかしWWWが一般に普及するとともに、文書構造が明確な論文スタイルの文書 ばかりでなく、カタログや広告のような、複雑な組み版処理などを表示する機能が求められるようになってきた。このためHTML仕様は、バージョンアップするたびに、このような表示機能を追加してきたという背景がある。

CSSの基本的な思想は、文書とレイアウト定義を別々に規定可能にし、その文書を表示するユーザーが適当なレイアウトを設定できるようにすることだが、現在のブラウジングソフトでも完全に組み込まれているのはCSS2の一部の機能だけで、このようにユーザーが任意のレイアウト設定を選択したりすることはできない。

(http://www.ascii.co.jp/pb/ghelp/28/002801.htmlより抜粋、一部修正)

4. DOM (Document Object Model)

1) Webページ中の文字や画像などの要素をオブジェクトとして扱い、その属性やメソッドを利用することで、Webページを動的に変化させることを目的としたテクノロジ。DHTMLの主要な機能の1つ。DHTMLでは、すべてのオブジェクト上で、キーボード入力やマウスクリックなどのイベントを利用することが可能である。

DOMによって、以下の3つの機能が実現される。

Dynamic Content

文字や画像を、動的に挿入したり削除したりするための機能。そのほか、文字列の 選択や検索なども行える。

Dynamic Styles

Internet Explorer 4.0では、Cascading Style Sheet (CSS) をサポートしているが、Dynamic Styles機能を使用すれば、そのスタイルの内容を動的に変更することができる。たとえば、マウスカーソルでポイントした文字列の、色やフォントサイズを変更するといったことが可能になる。

Absolute Positioning

Webページ中のオブジェクトを、ページの任意の位置に表示するための機能。この機能も、ページ表示後に動的に利用することができるので、スクリプティングと組み合わせれば簡単なゲーム程度を作成できる。

2) プログラムやスクリプトを利用して、Webページのコンテンツや構造、ドキュメントスタイルにダイナミックにアクセスしたり、更新したりできるようにするためのオブジェクトインターフェイス。W3Cのワーキングループにより考案され、'97年10月に最初のドラフト仕様が公開された。

(http://www.ascii.co.jp/pb/ghelp/34/003435.html)

5. GUI (graphical user interface)

伝統的な文字ベースのインターフェイスではなく、ボタンやメニューなどのグラフィックスの部品を使って設計されたユーザーインターフェイス。Graphical User Interfaceの頭文字を使ってGUIと呼ばれることも多い。(「GUI」は「ジー・ユー・アイ」または「グイ」と読む)。(http://www.ascii.co.jp/pb/ghelp/5/000598.html)

6. HTML (Hyper Text Markup Language)

「エイチ ティ エム エル」と読む。

HTMLはSGML(Standard Generalized Markup Language)の書式を踏襲したマークアップ言語の1つ。HTMLはWWWサーバでのドキュメントを記述するための言語として広く知られている。WWWサーバのドキュメントでは、このHTMLにより文書の構造や体裁などの要素(SGMLやHTMLでは、これらをエレメントと呼んでいる)を定義し、イメージやURLを貼り込んで、ハイパテキストを実現している。

(http://www.ascii.co.jp/pb/ghelp/13/001364.html)

7. PDF (Portable Document Format)

Portable Document Format(PDF)とは、米国Adobe Systems社がデジタル書類によるコミュニケーションを実現するために開発したファイル形式のことで、書類に含まれるあらゆる内容(文字、画像、レイアウト情報など)が、「Acrobat Reader」を使用することによって表示・出力できる電子文書配布用のデータ形式である。

(http://www.dnp.co.jp/pdf-w/)

8. RDF (Resource Description Framework)

WWWを利用するアプリケーション間での効果的なデータ交換を可能にするためのメタデータフレームワーク。RDFでは、メタデータのプロパティを定義可能にし、異なるアプリケーション同士が、WWWを通じてデータの交換を行えるようにする。このプロパティセットを利用することで、異なるアプリケーション間で人間が判別可能な状態でデータを交換したり、ソフトウェアが解釈可能な形でデータを交換したりできるようになる。

現在RDFは、W3CのRDFワーキンググループにおいて仕様案の策定が進んでいる。 W3Cの概要説明によれば、サイトマップやレーティング、ストリームチャンネルの定 義、サーチエンジン、認証の発行などでRDFの技術が活かされるとしている。

(http://www.ascii.co.jp/pb/ghelp/34/003431.html)

9. SGML (Standard Generalized Markup Language)

1986年にISOで制定(ISO8879-1986)された電子的に文書を取り扱うための計算機言語。DTD(Document Type Definition;文書型定義)の1つ。1992年に日本工業規格になる(JIS X4151)。「文書記述言語 S G M L 」として制定。文書を構造化するための基本セットである。HTMLはこのSGMLの応用利用(アプリケーション)に位置する。

10. T-Time

TTimeは、プレーンテキスト、HTMLなどのテキストデータに対して手軽に迅速に好みの文字面を設定。モニタの上でそのまま紙みたいにスイスイ読むための「テキストビューワー」である。(http://www.voyager.co.jp/T-Time/)

11. Voice Explorer

大阪府立盲学校の横田氏により開発され、配布されているホームページを音声で読み上げるソフト。Internet Explorerの機能を利用して読み上げることができる。読み上げにはProTalkerが必要である。

12. W3C (World Wide Web Consortium)

WWW全般の標準化を行う団体。一般にはW3Cと略記されることが多い。W3Cは IETF(Internet Engineering Task Force)と呼ばれる委員会のワークグループで、WWWに関する具体的な標準化項目についてここで議論される。このうちHTMLは、IETFのApplication Areaと呼ばれるワークグループで標準化作業が進められている。W3C は 1994 年 7 月 に MIT と CERN に よって 設立 され、 現在 は、 MIT の LCS(Laboratory for Computer Science)とフランスのINRIA(The French National Institute for Research in Computer Science and Control)が中心となって運営されている。(http://www.ascii.co.jp/pb/ghelp/22/002201.html)

13. WWW (World Wide Web)

欧州核物理学研究所(European Center for Nuclear Research. CERN)で開発されたドキュメントシステム。ドキュメント中に他のドキュメントへのジャンプコマンドを埋め込むことができる。このジャンプコマンドとしては、URL(Uniform Resource Locator)を使用し、Internetに参加している世界中のWWWサーバのどのドキュメントにもジャンプすることができる。

WWWでドキュメントを提供するシステムはWWWサーバと呼ばれる(これに対し、クライアント側でWWWサーバのドキュメントをブラウズするソフトウェアはWWWクライアントと呼ばれる)。ドキュメントの記述には、HTML(HyperText Markup Language)と呼ばれるマークアップ言語を使用する。あるHTML文書と他のHTML文書をリンクするには、リンク先のURLを文書中の文字列とともに記述する。WWWのドキュメントとしては、文字はもちろんのこと、静止画(GIFファイルなど)や音声、

動画など、あらゆるデータを提供することが可能である。

HTML文書の解釈はブラウザ(WWWクライアント)によって行なわれ、HTML文書中に記述されたタグに基づいて表示がなされる。たとえば見出し文字の太さや大きさは、見出し行のレベルという形でタグ中に記述される。見出し行のレベルをどのように解釈して表示するかは、ブラウザに任されている。したがってブラウザが異なれば、表示も異なる可能性がある。(http://www.ascii.co.jp/pb/ghelp/9/000992.html)

14. XML (eXtensible Markup Language)

Internet上でのSGMLの利用を容易にすることを目的として設計されたマークアップ言語。HTMLのような固定のマークアップ方法だけではなく、文書独自のマークアップ方法を定義できるようになる。このXMLの標準化は、W3Cで行なわれている。

(http://www.ascii.co.jp/pb/ghelp/31/003150.html)

15. XSL (eXtensible Stylesheet Language)

Microsoft、ArborText、Insoによって提唱された拡張スタイルシート仕様。この XSLは、'97年9月にW3Cによって仕様が認定された。

XSLは、Webページのデータを表示する際のフォーマットを決定するスタイルシート (CSS: Cascading Style Sheet) の機能を拡張し、構造化されたXMLデータから、その表示順序を変更したり、フィルタリングしたりして、ユーザーがデータをブラウズできるようにする。これによりユーザーは、XMLデータの記述順序とは無関係に、独自のフォーマットでデータを表示できるようになる。

(http://www.ascii.co.jp/pb/ghelp/34/003430.html)

16. アンカー文字

ハイパリンクを定義づけるために、HTMLで定義されているタグにより構造化された文字。アンカー文字をクリックするか、フォーカスを合わせてエンターキーを押すことでことで、リンク先のファイルやその中の特定の文章を表示することができる。

17. インタプリタ

インタプリタ言語というのは、プログラムを処理するときに機械語に翻訳しながらすすめるような言語である。代表例として、JavaScriptはもちろん、BASICなどが挙げられる。インタプリタは、面倒な翻訳作業(コンパイル)がなく、さらに機械に直接命令するわけではないので、プログラムコードを全く修正することなく様々な種類のマシンで利用することができるという利点がある。

(http://sais.u-shizuoka-ken.ac.jp/~gakucom/detail/js/js_01.html)

18. コントラスト(contrast)

コントラストの算出法には、何種類かあるが、ここではModulation Contrastを用いる。公式は、Cm=(Lc·Lb)/(Lc+Lb)。(Lc;文字輝度、Lb;背景輝度)

19. 字詰まり効果(Crowding effect)

視対象が空間的に互いに接近していると認知成績が低下する現象。幼児や老人、ロービジョンで顕著な現象として知られている。(川嶋ら;1998)

20. ブラウザソフト

HTML文書を表示するためのソフト。Microsoft社とNetscape社のソフトがシェアを2分している。

21. ホームページリーダ

IBMから発売されている、ホームページを音声で読み上げるソフト。音声化するためには同社のProTalkerという音声ソフトが必要である。

22. ロービジョン(low-vision)

いろいろな病気の結果、治療することができず、また、メガネなどでも直せない目の機能低下によって、社会生活や教育で特別な配慮を必要とする、社会的・教育的な意味での弱視のこと。また、医学的弱視(amblyopia)は眼球や視神経に疾患がないのに視力が出ない状態をいう。(鳥居修晃, 視覚障害と認知, 放送大学教育振興会, 1993)

8 引用・参考文献

- ◆ Gordon E. Legge, Denis G. Pelli, Gary S. Rubin and Mary M. Schleske (1985a) PSYCHOPHYSICS OF READING-I. NORMAM VISION, Vision Res., 25(2), 239-252.
- ◆ Legge.G.E., Rubin.G.S., Pelli.D.G. Schleske.M.M (1985b) Psychophysics of reading-II Low vision, Vision Research, 25(2), 253-266.
- ◆ Legge, G.E., Rubin, G.S., and Schleske, M.M. (1987) Contrast polarity effects in low vision reading., in Woo.(ed), Low vision principles and application, Springer-Verlag, 288-307.
- ◆ 東陽一 (1999) PDFとXML/なにが違う・どう使う, OPENDESIGN, 35, 116-119.
- ◆ 新井千賀子・中村仁美・小田浩一・千田耕基(1999) 白子症と先天無虹彩における羞明とコントラストポラリティ効果-MNREAD-J,JKによる評価-, 第8回視 覚障害リハビリテーション研究発表大会論文集, 57-60.
- ◆ 氏間和仁 (1998) 学校教育におけるHTMLの活用,愛媛県高等学校教育研究会障 害児教育部会誌, 20, 12-21.
- ◆ 大川原潔ら (1992) 弱視児に対する拡大教材の必要性と望ましい教材拡大のあ り方、日本弱視教育研究会, 9.
- ◆ 大川原潔ら (1999) 視力の弱い子どもの理解と支援, 教育出版株式会社(東京)
- ◆ 岡崎泰久 (1999) 技術解説HTMLドキュメント作成技術とその教育利用, 教育 システム情報学会誌, 16(2), 124·125.
- ◆ 岡田伸一 (1990) VDT作業の弱視者への影響について -自覚症状調査の結果から-, 第3回ロービジョン研究発表大会論文集, 37-40.
- ◆ 小田浩一・J.Stephen Mansfied, Gordon E. Legge (1998) ロービジョンエイド を処方するための新しい読書検査表MNREAD·J, 第7回視覚障害リハビリテーション研究発表大会論文集, 157·160.
- ◆ 小田浩一・江坂百合子・中野泰志 (1993) フォントの見やすさ―視力低下がある場合、標準的な3つの書体はどれが一番読みやすいか?―, 第2回視覚障害リハビリテーション研究発表大会論文集, 50-53.
- ◆ 小野祐史 (1998) 紙から電子へ, 情報処理, 39(6), 507-513.
- ◆ 加藤博之・水野升裕 (1999), 情報の電子化技術 入門から応用まで 9

SGML/XML作成技法,情報管理,42(9),777-789.

- ◆ 神立景子 (1997) からだにやさしいPCの使い方, オーム社.
- ◆ 金城悟・中田英雄・佐藤泰正 (1989) CRT Displayのコントラストが視覚的探索に及ぼす影響, 第2回ロービジョン研究大会論文集, 21-26.
- ◆ 川嶋英嗣・小田浩一 (1998) 字詰まり効果と読書困難, 第7回視覚障害リハビリ テーション研究発表大会論文集, 125·128.
- ◆ 川嶋英嗣・小田浩一・藤田京子・中村仁美・香川邦生 (1999) 字詰まり効果と 読書困難(2). 第8回リハビリテーション研究発表大会論文集, 101-104.
- ◆ 菊地智明・中野泰志 (1992) 弱視者の読みに及ぼす文字サイズと行間隔の効果, 日本特殊教育学会第30回大会発表論文集,44-45.
- ◆ 菊地智明・中野泰志 (1993) 読書効率に及ぼす一行の文字数の効果, 日本特殊 教育学会第31回大会発表論文集, 78·79.
- ◆ 窪田悟・大倉元宏 (1985) VDTの文字表示極性と適正コントラストに関する実験的研究, 第1回ヒューマン・インターフェイス・シンポジウム論文集, 229-234.
- ◆ 佐藤守・中野泰志 (1993) PCを利用した読みの指導, 弱視教育, 31(1), 7-13.
- ◆ 島田純子・飯島邦夫・浅野光基・森田歌子 (1999) 情報の電子化技法-入門から応用まで第3回ディジタルコンテンツの活用事例─HTML─『情報管理』誌を例として、情報管理、42(3)、246·256.
- ◆ 鈴木亨・秋谷忍・斉藤進 (1993) VDT画面注視時の眼球回転角, 臨床眼科, 47(8), 1519-1522.
- ◆ 総理府, 障害者白書平成10年版, 1998.
- ◆ 外山みどり・斉藤真・Taptagaporn S (1992) 視線方向を考慮した快適VDTワークステーションの設計, 第8回ヒューマン・インターフェイス・シンポジウム 論文集, 517-522.
- ◆ 高橋尚子・池谷尚剛・中田英雄・谷村裕(1990) フリッカー値から見た低視力者の視疲労, 第3回ロービジョン研究発表大会論文集, 41-47.
- ◆ 坪田一男・八木幸子・戸田郁子 (1992) ドライアイ発症における瞬目回数と眼 験列幅の重要性、日本眼科学会雑誌 96 臨時創刊, 227.
- ◆ 鳥居修晃 (1993) 視覚障害と認知, 放送大学教育振興会.
- ◆ 中野泰志 (1991) 弱視者の視認性を考慮した文字の効果的な提示方法(1) コンピュータディスプレイでの白黒反転効果,電子情報通信学会技術研究報告,91(315),15·22.

- ◆ 中野喜美子・中野泰志 (1992) 読書効率に及ぼす文字サイズの効果 人工的視力低下状態での最適文字サイズの検討,日本特殊教育学会 第30回大会発表論文集,14-15.
- ◆ 中野泰志・佐藤守・菊地智明(1993) 行たどりに困難を示す弱視児のためのコンピュータを用いた新しい読書補助具の試作,国立特殊教育総合研究所研究紀要,第20巻,89-96.
- ◆ 中野泰志・関みどり・菊地智明・高橋恵子・本田達也 (1994) PCを用いた弱視 用リーディングエイドの試作(2)―学習教材の電子化を考慮した改良―, 第3回 視覚障害リハビリテーション研究発表大会論文集, 120-123.
- ◆ 中野喜美子・中野泰志 (1995) 弱視児の読書環境の整備に関する事例報告・読書 効率の評価結果を通しての弱視児との関わり・, 弱視教育, 32(4), 10·16.
- ◆ 葉石光一 (1995) 弱視児の読みに対する教育的配慮・心理学的研究からの示唆・, 弱視教育, 33(1), 15-21.
- ◆ ビレッジセンターHTML&SGML研究チーム(1999),正しいHTML4.0リファレンス&作法,株式会社ビレッジセンター出版局.
- ◆ 古田信子・青木成美 (1989) 弱視児の見え方に及ぼす白黒反転の効果, 弱視教育, 27(2), 6·8.
- ◆ 保田明夫 (1999) 情報の電子化技法──入門から応用まで第2回 ディジタルコン テンツの作成技法-HTML-, 情報管理,42(2), 155-166.
- ◆ バーバラ・ブラウン著 簗島謙次監訳 (1999) ロービジョンハンドブック,診断 と治療社.
- ◆ 弓削経一(1966)幼年弱視,金原出版株式会社.
- ◆ IETF.HTML2.0 (RFC1866), (http://www.ietf.org/rfc/rfc1866.txt).
- ◆ W3C.HTML3.2 (REC-HTML32), (http://www.w3.org/TR/REC-HTML32/).
- ◆ W3c.HTML4.0 (REC-html40-19980424), (http://www.w3.org/TR/REC-html40/).
- ◆ References on Web Accessibility (http://www.w3.org/WAI/References/).
- ◆ すべての人にアクセス可能なHTML文書を書く、(http://www.twcu.ac.jp/~k-oda/AccessBlind/AccessibleHTML.html).

巻末 資料

巻末資料には、報告書中に出てきた表示ツールのソースコードと、実験で得られたデータ、実験で用いた読材料を収めています。また、本報告書にはCD-ROM版も用意しています。CD-ROMには、ここに載せることができなかった教材やスライドなどの資料も掲載しています。必要な方は問い合わせてください。

1	ソースコード	1
2	実験データ	24
3	読材料	26

1 ソースコード

23行目 </body>
24行目 </html>

行番号は、便宜上付けている。

index.htmlのソースコード

1行目 <html> 2行目 <head> <title>愛媛県立松山盲学校メインページ</title> 3行目 4行目 <script language="JavaScript" src="text.js"> 5行目 </script> 6行目 </head> 7行目 <body id="L1" onload="Effect()" onkeydown="KeyC(0)" ondblclick="Back()"> 8行目 <h1>愛媛県立松山盲学校
ホームページ</h1> 9行目 10行目 11行目 学校紹介 12行目 お勉強の部屋 13行目 <ii>> 児童・生徒の部屋 14行目 職員の部屋 15行目 学校からのお知らせ 16行目 その他 17行目 このページの使い方 18行目 19行目 20行目 地図 21行目 スペースで環境設定 22行目 ダブル・クリックで前のページへ戻れます。

text.jsのソースコード

```
<!--
1行
                                                 //フォントサイズ用
2行
              var N;
3行
              var FC:
                                                 //文字色用
4行
              var BC;
                                                 //背景色用
5行
                                                 //リンクテキスト用
              var LC;
6行
                                                 //訪問済みリンクテキスト
              var VC;
7行
                                                        //アクティブ文字色
              var AC;
                                                 //キーコードを保存
8行
              var KeyPushed;
9行
                                                 //行間隔
              var LH;
10行
              var LS;
                                                 //Later Spacing
11行
                                                 //URL
              var UL;
12行
       N=getCookie("Ns");
13行
14行
       if (N==""){}
15行
              N=30;
                                                 //フォントサイズ用
16行
              FC="white":
                                                 //文字色用
17行
              BC="black";
                                                 //背景色用
18行
              LC="white";
                                                 //リンクテキスト用
19行
              VC="white";
                                                 //訪問済みリンクテキスト用
20行
              AC="white":
                                                 //アクティブ文字色
21行
              LH=15;
                                                 //行間隔設定
22行
              LS=5:
                                                 //文字間隔設定
23行
                                                 //URL格納
              UL=document.URL;
24行
              varA=UL.length;
25行
              varA=varA-10;
26行
              UL=UL.substring(0,varA);
27行
              Save();
28行
             }
29行
30行
       UL=getCookie("ULs").replace(/%20/," ");
31行
       FC=getCookie("FCs");
32行
      var sCSS;
33行
34行
          if (FC=="black") {
35行
            sCSS = '<LINK REL="stylesheet" TYPE="text/css" HREF="' + UL + 'Bw.css">';
```

3

```
}
36行
37行
           else {
38行
               if(FC=="white"){
             sCSS = '<LINK REL="stylesheet" TYPE="text/css" HREF="' + UL + 'Wb.css">;
39行
40行
41行
               }
           document.write(sCSS);
42行
               window.status=sCSS;
43行
44行
45行
       function Effect(){
                                                     //フォントサイズ用
               N=getCookie("Ns");
46行
               FC=getCookie("FCs");
                                                     //文字色用
47行
                                                     //背景色用
48行
               BC=getCookie("BCs");
                                                     //リンクテキスト用
               LC=getCookie("LCs");
49行
                                                     //訪問済みリンクテキスト用
               VC=getCookie("VCs");
50行
                                                     //アクティブ文字色
51行
               AC=getCookie("ACs");
                                                     //行間隔
               LH=getCookie("LHs");
52行
                                                     //文字間隔
               LS=getCookie("LSs");
53行
54行
               LH=N*(LH/10);
               LS=LS/10;
55行
56行
57行
58行
          if(document.layers){
               document.layers("L1").style.letterSpacing=LS+"em";
59行
               document.layers("L1").style.lineHeight=LH+"pt";
60行
               document.layers("L1").style.fontSize=N+"pt";
61行
               document.bgColor=BC;
62行
               document.fgColor=FC;
63行
                document.linkColor=LC;
64行
                document.vlinkColor=VC;
65行
                document.alinkColor=AC;
66行
 67行
           }
 68行
 69行
           if(document.all){
                document.all.L1.style.letterSpacing=LS+"em";
 70行
                document.all.L1.style.lineHeight=LH+"pt";
 71行
                document.all.L1.style.fontSize=N+"pt";
 72行
 73行
                document.bgColor=BC;
                document.fgColor=FC;
 74行
```

```
100 OpenWinO
                                                                             示EII
                                                                             112/1
                                                                         {
                                                                             1111
                                               setCookie("ULs", UL);
                                                                             TYOLL
                                                setCookie("LSs", LS);
                                                                             하601
                                               setCookie("LHs", LH);
                                                                             7780 f
                                              setCookie("ACs", AC);
                                                                             차/01
                                              setCookie("VCs", VC);
                                                                             T1601
                                               setCookie("LCs", LC);
                                                                             計20f
                                              setCookie("BCs", BC);
                                                                             ₩104
                                               setCookie("FCs", FC);
                                                                             TIEOI
                                                 setCookie("Ns",N);
                                                                             計SOL
                                                           function Save(){
                                                                             하이
                                                                             ₹1000F
                                                                          {
                                                                              T}99
                                                                              T)89
                                           document.cookie=tmp;
                          tmp+="expires=Fri, 31-Dec-2030 23:59:59; ";
                                                                              T)79
                                     tmb=keγ+"="+escape(√αl)+"; ";
                                                                              779
。むびき書き燃変~ーキャイ||
                                               function setCookie(key,val){
                                                                              示69
                                                                              다146
                                                                          {
                                                                              示69
                                                                  {
                                                                              1729
                                                  return("");
                                                                              示19
                                                              else{
                                                                              1100
                                                                              11/98
                         return(unescape(tmp.substring(start,end)));
                                                                              形88
                                 end=tmp.indexOf(";",start);
                                                                              규/8
                                 ;[+(0,"=")}Oxebni.qmt=hota
                                                                               Tî 68
                      tmp=lam1,fqm1)gnintsdus.qm1=qm1;
                                                                               計28
                                                       }([-=|[dmt)ji
                                                                               予48
                                         tmp1=tmp.indexOf(key,0);
                                                                               T1 58
                                         tmp=document.cookie+";";
                                                                               T}28
                                                  function getCookie(key){
                                                                               計18
                                                                               TÎ08
                                                                          {
                                                                               7987
                                                                       {
                                                                               1787
                                         document.alinkColor=AC;
                                                                               TYY
                                                                               1767
                                          document.vlinkColor=VC;
                                           document.linkColor=LC;
                                                                               :ñ87
```

```
location.href = UL + "config1.html";
114行
115行
      }
116行
117行
       function KeyC(){
118行
              KeyPushed=(event.keyCode);
                                   //スペースで設定ページへ
119行
              if(KeyPushed=="32"){
              OpenWin();
120行
121行
              }
122行
      }
123行
124行
      function Back(){
125行
              history.back()
126行
              }
127行 //-->
```

config1.htmlのソースコード

```
1行目
      <html>
2行目
      <head>
      <title>設定のページ1 (配色の設定) </title>
3行目
4行目
      <script language="JavaScript">
5行目
      <!--
             var N; //フォントサイズ用
6行目
7行目
             var FC: //文字色用
8行目
             var BC; //背景色用
             var LC; //リンクテキスト用
9行目
             var VC; //訪問済みリンクテキスト用
10行目
11行目
             var AC; //active charactor color
             var LH; //行間隔用
12行目
             var LS; //文字間隔用
13行目
14行目
15行目
             BC=getCookie("BCs");
16行目
             FC=getCookie("FCs");
17行目
             LC=getCookie("LCs");
18行目
             VC=getCookie("VCs");
19行目
             AC=getCookie("ACs");
20行目
             N=getCookie("Ns");
21行目
             LH=getCookie("LHs");
```

```
}([-=![qmt])i
                                                                         目示86
                                         tmp1=tmp.indexOf(key,0);
                                                                         目示78
                                                                         目示68
                                        tmp=document.cookie+";";
                                                 25行日 function getCookie(key){
                                                                         目示42
                                                                       { 目示62
                                               setCookie("LSs", LS);
                                                                         目示公司
                                             setCookie("ACs", AC);
                                                                         目示ら
                                                                         目示08
                                               setCookie("LHs",LH);
                                             setCookie("VCs", VC);
                                                                         目刊94
                                                                         目示84
                                              setCookie("LCs", LC);
                                              setCookie("BCs", BC);
                                                                         目示74
                                              setCookie("FCs", FC);
                                                                         月示24
                                                setCookie("Ns",N);
                                                                          目1764
                                                         function Save(){
                                                                         目介44
                                                                          目示54
                                                                       { 目示24
                                   document, all, L1, innerHTML=tmp;
                                                                          目引作
                                                                          目示04
                                                                          目示96
                                                         01*S7=S7
                                                                          日示86
                                                      :01*N/HJ=HJ
                                                                          目示75
                                         document.alinkColor=AC;
                                         document.vlinkColor=VC;
                                                                          目示66
                                                                          日計68
                                           document.linkColor=LC;
                                                                          目計48
                                            document.fgColor=FC;
                                           document.bgColor=BC;
                                                                          目示68
                               document, all, L1, style, fontSize=N=musob
                                                                          32行目
                         document.all.L1.style.letterSpacing=L5+"em";
                                                                          目計16
                             document.all.L1.style.lineHeight=LH+"pt";
                                                                          目示の8
                                                                   :,,<d/>
「殊全づ1etne ,む逝づ← ,る氣ゔ→<1d>更変つくや氷↓↑<q><1d><q\><1d><1d><2</p>
('<\mtn.fginoc'=fenter>配合('<\mthrace)>家级()>家级()
                                                                          13.65
                                                                          目 計82
                                                                          目77亿目
                                                         :01/S7=S7
                                                     (01/HJ)*N=HJ
                                                                          目示62
                                                                          月7762
                                                         function Effect(){
                                                                          目行24
                                                                          月755
                                               LS=getCookie("LSs");
                                                                          日小公
```

```
59行日
                      tmp=tmp.substring(tmp1,tmp.length);
60行目
                      start=tmp.indexOf("=",0)+1;
61行目
                      end=tmp.indexOf(";",start);
                      return(unescape(tmp.substring(start,end)));
62行目
63行目
              }
64行目
              else{
65行目
                      return("");
66行目
              }
67行目 }
68行目
69行目 function setCookie(key,val){
70行目
              tmp=key+"="+escape(val)+"; ";
71行日
              tmp+="expires=Fri, 31-Dec-2030 23:59:59; ";
72行目
              document.cookie=tmp;
73行目 }
74行目
75行目 function BWO{
76行目
              FC="black";
77行目
              BC="white";
78行目
              LC="#000000";
79行目
              VC="#000000";
80行目
              AC="#000000";
81行目
              Effect();
82行目 }
83行目 function WBO{
84行目
              FC="white";
               BC="black";
85行日
               LC="#ffffff";
86行目
87行目
               VC="#ffffff";
               AC="#ffffff";
88行目
89行目
               Effect();
90行目 }
91行目
92行目 function KeyCO{
93行目
               KeyPushed=(event.keyCode);
                                            //
                                            //↑で黒地に白文字
94行目
               if(KeyPushed=="38"){
95行目
               BWO;
96行目
               }
                                            //↓で白地に黒文字
97行目
               if(KeyPushed=="40"){
```

```
98行目
              WB();
99行目
              }
              if(KeyPushed=="37"){
                                          //←で黒地に白文字
100行目
              Back();
101行目
102行目
              }
                                           //→で白地に黒文字
              if(KeyPushed=="39"){
103行目
              location.href="config2.html";
104行目
105行目
              Save();
106行目
              }
                                           //enterで保存・終了
              if(KeyPushed=="13"){
107行目
108行目
              BackSave();
109行目
              }
110行目 }
111行目
112行目
              function Back(){
113行目
              history.back();
114行目
              Save();
115行目
              }
116行目
              function BackSave(){
117行目
              Save();
118行目
              history.back();
119行目
              }
120行目
121行目 //-->
122行目 </script>
123行目
124行目 </head>
125行目 <body id="L1" onload="Effect()" onkeydown="KeyC()">
126行目 </body>
127行目 </html>
```

config2.html \mathcal{O} \mathcal{V} \mathcal{A} \mathcal{A} \mathcal{A} \mathcal{A} \mathcal{A}

```
1行目 <html>
2行目 <head>
3行目 <title>設定のページ 2 (文字サイズの設定) </title>
4行目 <script language="JavaScript">
5行目 <!--
```

```
6行目
                           //フォントサイズ用
             var N;
7行目
             var FC;
                           //文字色用
                           //背景色用
8行目
             var BC:
                           //リンクテキスト用
9行目
             var LC;
                           //訪問済みリンクテキスト用
10行目
             var VC;
11行目
             var AC:
                           //active charactor color
12行目
             var LH;
                           //行間隔用
13行目
             var LS:
                           //文字間隔用
14行目
15行目
             BC=getCookie("BCs");
16行目
             FC=getCookie("FCs");
             LC=getCookie("LCs");
17行目
18行目
             VC=getCookie("VCs");
19行目
             AC=getCookie("ACs");
20行目
             N=getCookie("Ns");
21行目
             LH=getCookie("LHs");
             LS=getCookie("LSs");
22行目
23行目
24行目 function Effect(){
25行目
26行目
             LH=N*(LH/10);
27行日
             LS=LS/10;
28行日
29行目 tmp= "<align=center> 文字サイズの設定 </a><br>"+N+"ポイントです。
       <hr>↑ ↓ボタンで変更<br>←で戻る、→で進む、enterで全終了";
30行目
             document.all.L1.style.lineHeight=LH+"pt";
             document.all.L1.style.letterSpacing=LS+"em";
31行目
32行目
              document.all.L1.style.fontSize=N+"pt";
33行目
              document.bgColor=BC;
34行目
             document.fgColor=FC;
35行目
             document.linkColor=LC;
              document.vlinkColor=VC;
36行目
              document.alinkColor=AC;
37行目
38行目
              LH=LH/N*10;
39行目
              LS=LS*10;
40行目
41行目
              document.all.L1.innerHTML=tmp;
42行目 }
43行目
```

```
44行目 function Save(){
45行日
               setCookie("Ns",N);
46行目
               setCookie("FCs",FC);
47行目
               setCookie("BCs",BC);
               setCookie("LCs",LC);
48行目
49行目
               setCookie("VCs", VC);
               setCookie("LHs",LH);
50行目
51行目
               setCookie("ACs",AC);
               setCookie("LSs",LS);
52行日
53行目 }
54行目
55行目 function getCookie(key){
               tmp=document.cookie+";";
56行目
57行目
               tmp1=tmp.indexOf(key,0);
58行目
               if(tmp1!=-1){
                       tmp=tmp.substring(tmp1,tmp.length);
59行目
                       start=tmp.indexOf("=",0)+1;
60行目
                       end=tmp.indexOf(";",start);
61行目
                       return(unescape(tmp.substring(start,end)));
62行目
               }
63行目
64行目
               else{
65行目
                       return("");
               }
66行目
67行目 }
68行目
69行目 function setCookie(key,val){
70行目
               tmp=key+"="+escape(val)+"; ";
71行目
               tmp+="expires=Fri, 31-Dec-2030 23:59:59; ";
               document.cookie=tmp;
72行目
73行目 }
74行目
75行目 function Big(){
76行目
               N=N-1;
77行目
               N=N+2;
78行目
               if (N>100){N=100};
79行目
               Effect();
80行目 }
81行目 function SmallO{
82行目
               N=N-1:
```

```
if (N<6) \{N=6\};
83行目
              Effect();
84行目
85行目 }
86行目
87行目
88行目 function KeyCO{
              KeyPushed=(event.keyCode);
89行目
                                          //↑で黒地に白文字
90行目
              if(KeyPushed=="38"){
91行目
              Big();
92行目
              }
                                          //↓で白地に黒文字
93行目
              if(KeyPushed=="40"){
              Small();
94行目
95行目
              }
                                           //←で黒地に白文字
96行目
              if(KeyPushed=="37"){
97行目
              Back();
98行目
              }
                                           //→で白地に黒文字
              if(KeyPushed=="39"){
99行目
              location.href="config3.html";
100行目
101行目
              Save();
102行目
              }
                                           //enterで保存・終了
103行目
              if(KeyPushed=="13"){
104行目
              BackSave();
105行目
106行目 }
107行目
              function Back(){
108行目
109行目
              history.back();
110行目
              Save();
111行目
              function BackSave(){
112行目
113行目
              Save();
              history.go(-2);
114行目
115行目
              }
116行目
 117行目 //-->
 118行目 </script>
 119行目
 120行目 </head>
 121行目 <body id="L1" onload="Effect()" onkeydown="KeyC()">
```

122行目 </body> 123行目 </html>

config3.htmlのソースコード

```
1行目
      <html>
2行目
      <head>
      <title>設定のページ3 (行間隔の設定) </title>
3行目
4行目
      <script language="JavaScript">
5行目
      <!--
                           //フォントサイズ用
6行目
             var N;
7行目
             var FC;
                           //文字色用
8行目
                           //背景色用
             var BC;
9行目
             var LC;
                           //リンクテキスト用
10行目
             var VC;
                           //訪問済みリンクテキスト用
11行目
                           //active charactor color
             var AC;
12行目
             var LH;
                           //行間隔用
13行目
             var LS;
                           //文字間隔用
14行目
15行目
             BC=getCookie("BCs");
16行目
             FC=getCookie("FCs");
17行目
             LC=getCookie("LCs");
18行目
             VC=getCookie("VCs");
19行目
             AC=getCookie("ACs");
20行目
             N=getCookie("Ns");
21行目
             LH=getCookie("LHs");
22行目
             LS=getCookie("LSs");
23行目
24行目
      function Effect(){
25行目
             LH=N*(LH/10);
26行目
             LS=LS/10;
27行目
28行目 tmp= "<align=center>行間隔の設定</a><br>"+LH/N+"行です。<hr>↑↓
      ボタンで変更<br>
←で戻る、→で進む、enterで全終了";
29行目
             document.all.L1.style.lineHeight=LH+"pt";
30行目
             document.all.L1.style.letterSpacing=LS+"em";
31行目
             document.all.L1.style.fontSize=N+"pt";
32行目
             document.bgColor=BC;
```

```
33行目
               document.fgColor=FC;
34行目
               document.linkColor=LC:
35行目
               document.vlinkColor=VC:
               document.alinkColor=AC:
36行目
37行目
               LH=LH/N*10:
               LS=LS*10;
38行目
39行目
40行目
               document.all.L1.innerHTML=tmp;
41行目 }
42行目
43行目 function Save(){
44行目
               setCookie("Ns",N);
4577 H
               setCookie("FCs",FC);
46行目
               setCookie("BCs",BC);
47行目
               setCookie("LCs",LC);
               setCookie("VCs",VC);
48行目
49行目
               setCookie("LHs",LH);
50行目
               setCookie("ACs",AC);
               setCookie("LSs",LS);
51行目
52行目 }
53行目
54行目 function getCookie(key){
55行目
               tmp=document.cookie+";";
56行目
               tmp1=tmp.indexOf(key,0);
57行目
               if(tmp1!=-1){
58行目
                      tmp=tmp.substring(tmp1,tmp.length);
59行目
                      start=tmp.indexOf("=",0)+1;
60行目
                      end=tmp.indexOf(";",start);
                      return(unescape(tmp.substring(start,end)));
61行目
62行目
               }
63行目
               else{
64行目
                      return("");
65行目
               }
66行目 }
67行目
68行目 function setCookie(key,val){
               tmp=key+"="+escape(val)+"; ";
69行目
70行目
               tmp+="expires=Fri, 31-Dec-2030 23:59:59; ";
71行目
               document.cookie=tmp;
```

14 卷末資料

```
72行目 }
73行目
74行目 function LHup(){
75行日
              LH=LH+5;
76行目
              if(LH>30)\{LH=30\}
77行目
              Effect();
78行目 }
79行目 function LHdn(){
80行目
              LH=LH-5;
81行目
              if(LH <= 10)\{LH = 10\}
82行目
              Effect();
83行目 }
84行目 function KeyC(){
85行目
              KeyPushed=(event.keyCode);
                                           //
              if(KeyPushed=="38"){
86行目
                                           //↑で黒地に白文字
87行目
              LHup();
88行目
              }
89行目
              if(KeyPushed=="40"){
                                           //↓で白地に黒文字
90行目
              LHdn();
91行目
              }
92行目
              if(KeyPushed=="37"){
                                           //←で黒地に白文字
93行目
              Back();
94行目
              }
95行目
              if(KeyPushed=="39"){
                                           //→で白地に黒文字
96行目
              location.href="config4.html";
              Save();
97行目
98行目
              }
              if(KeyPushed=="13"){
                                           //enterで保存・終了
99行目
100行目
              BackSave();
101行目
              }
102行目 }
103行目
104行目
              function Back(){
105行目
              history.back();
106行目
              Save();
107行目
              }
108行目
              function BackSave(){
109行目
              Save();
110行目
              history.go(-3);
```

```
111行目 }
112行目
113行目 //-->
114行目 </script>
115行目
116行目 </head>
117行目 <body id="L1" onload="Effect()" onkeydown="KeyC()">
118行目 </body>
119行目 </html>
```

config4.htmlのソースコード

```
1行目
      <html>
2行目
      <head>
      <title>設定のページ4 (文字間隔の設定) </title>
3行目
4行目
      <script language="JavaScript">
5行目
      <!--
6行目
             var N; //フォントサイズ用
             var FC; //文字色用
7行目
             var BC; //背景色用
8行目
             var LC; //リンクテキスト用
9行目
             var VC; //訪問済みリンクテキスト用
10行目
             var AC; //active charactor color
11行目
             var LH; //行間隔用
12行目
             var LS; //文字間隔用
13行目
14行目
             BC=getCookie("BCs");
15行目
16行目
             FC=getCookie("FCs");
             LC=getCookie("LCs");
17行目
18行目
             VC=getCookie("VCs");
             AC=getCookie("ACs");
19行目
20行目
             N=getCookie("Ns");
             LH=getCookie("LHs");
21行目
22行目
             LS=getCookie("LSs");
23行目
24行目 function Effect(){
25行目
26行目
              LH=N*(LH/10);
```

```
else{
                                                                          目示48
                                                                          目示63
                  return(unescape(tmp.substring(start,end)));
                                                                          目示28
                                  :(hpts,";")oxebni.qmt=bne
                                                                          目示19
                                 :[+(0,"=")1Oxebni.qmt=thats
                                                                          目示09
                        tmp=tmp.supstring(tmp1,tmp.length);
                                                                          目示98
                                                      }([-=![qmt])i
                                                                          目示88
                                         tmp1=tmp.indexOf(key,0);
                                                                          目示78
                                         tmp=document.cookie+";";
                                                                          目示99
                                                  function getCookie(key){
                                                                          目示68
                                                                          目示48
                                                                        { 目诉63
                                                setCookie("LSs", LS);
                                                                          目於SB
                                              setCookie("ACs", AC);
                                                                          目刊18
                                                setCookie("LHs", LH);
                                                                          目示08
                                              setCookie("VCs", VC);
                                                                          目示94
                                               setCookie("LCs", LC);
                                                                          目示84
                                               setCookie("BCs", BC);
                                                                          目示74
                                               setCookie("FCs",FC);
                                                                          目示24
                                                 setCookie("Ns",N);
                                                                          目示24
                                                          function Save(){
                                                                          目介44
                                                                           目示54
                                                                        { 目示24
                                    document.all.L1.innerHTML=tmp;
                                                                           目引作
                                                                           目示04
                                                                           39行目
                                                         :01*S1=S1
                                                                           目示85
                                                      (01*N/H1=H1
                                                                           目示75
                                         document.alinkColor=AC;
                                                                           目示66
                                          document.vlinkColor=VC;
                                                                           目示68
                                           document.linkColor=LC;
                                            :D7=10loDgj.tnemuoob
                                                                           34行目
                                           document.bgColor=BC;
                                                                           目示68
                                                                           32行目
                               document.all.L1.style.fontSize=H+"pt";
                         document.all.L 1.style.letterSpacing=L5+"em";
                                                                           31行目
                             document.all.L1.style.lineHeight=LH+"pt";
                                                                           目示05
                              "<q\>「豨全⑦19tn9、る気⑦→<1d>更変⑦くを氷
↓ ↑ <q><1d><q\>。マママ文"+2J+"<q><1d><b/>>京號の副間字文<1etneo=ngilo>" =qmt
                                                                           目示95
                                                                           目示82
                                                                           目示亿
                                                         :01/S7=S7
```

```
65行目
                      return("");
66行目
              }
67行目 }
68行目
69行目 function setCookie(key,val){
70行目
               tmp=key+"="+escape(val)+"; ";
71行目
               tmp+="expires=Fri, 31-Dec-2030 23:59:59; ";
               document.cookie=tmp;
72行目
73行目 }
74行目
75行目 function LSup(){
               LS=LS+5;
76行目
77行目
               if(LS>20)\{LS=20\}
78行目
               Effect();
79行目 }
80行目 function LSdn(){
81行目
               LS=LS-5;
82行目
               if(LS <= 0)\{LS = 0\}
83行目
               Effect();
84行目 }
85行目
86行目 function KeyCO{
               KeyPushed=(event.keyCode);
87行目
88行目
               if(KeyPushed=="38"){
89行目
               LSup();
90行目
               }
               if(KeyPushed=="40"){
91行目
92行目
               LSdn();
93行目
94行目
               if(KeyPushed=="37"){
95行目
               Back();
96行目
               }
97行目
               if(KeyPushed=="13"){
98行目
               BackSave();
99行目
               }
100行目 }
101行目
102行目
               function Back(){
103行目
               history.back();
```

```
Save();
104行目
105行目
              }
106行目
              function BackSave(){
107行目
              Save();
108行目
              history.go(-4);
109行目
              }
110行目
111行目 //-->
112行目 </script>
113行目
114行目 </head>
115行目 <body id="L1" onload="Effect()" onkeydown="KeyC()">
116行目 </body>
117行目 </html>
```

Wb.cssの ソースコード

```
1行目
       <!--
2行目
       H1{
               font-size:1.5em;
3行目
               margin-top:1.5em;
4行目
               margin-bottom:1.5em;
5行目
               text-align:center;
6行目
               line-height:1.3em;
7行目
               }
8行目
9行目
       H1:before{display:inline;
10行目
               counter-increment: "level1";
               content:counter("level1")"章 ";
11行目
               counter-reset: "level2";
12行目
13行目
               }
14行目
15行目 H2{
               font-size:1.3em;
16行目
               margin-top:1em;
17行目
               margin-bottom:1em;
18行目
               text-align:center;
19行目
               line-height:1.3em;
20行目
               }
21行目 H2:before{display: inline;
```

```
counter-increment:"level2";
22行目
              content:"第"counter("level2")"節
23行目
24行目
              content:"▼";
25行目
              }
26行目 H1 + H2{margin-top:0em}
27行目
28行目 H3{
              font-size:1.2em;
29行目
              margin-top:1em;
30行目
              margin-bottom:0.5em;
31行目
              text-align:left;
32行目
              line-height:1.3;
33行目
              margin-left:2em;
34行目 }
35行目 H2 + H3{margine-top:0em}
36行目
37行目 H4{
              font-size:1.1em:
38行目
              margin-top:lem;
39行目
              margin-bottom:0.5em;
40行目
              text-align:left;
41行目
              line-height:1.3;
42行目
               margin-left:1.5em;
43行目 }
44行目 H3 + H4{margine-top:0em}
45行目
46行目 H5{
               font-size:1em;
47行目
               margin-top:0.5em;
48行目
               margin-bottom:0.5em;
49行目
               text-align:left;
50行目
               line-height:1.2;
51行目
               margin-left:1em;
52行目 }
53行目
       H4 + H5{margine-top:0em}
54行目
55行目 P{
               text-indent:1em}
56行目 p#idt01{margin-left:1em}
57行目 p#idt02{margin-left:2em}
58行目 p#idt03{margin-left:3em}
59行目 p#idt04{margin-left:4em}
60行日 p#idt05{margin-left:5em}
```

```
61行目 p#idt06{margin-left:6em}
62行目
63行目 LI{
              margin-left:1em;
64行目
65行目 table{ font-size:1em;
66行目
              border-style:outset;
67行目
              margin-top:1.5em;
68行目
              margin-bottom:1.5em;
69行目
70行目 th,td{ border-style:outset;
71行目
              }
72行目
73行目
74行目 sup, sub{font-size:90%;
75行目
              }
76行目
77行目 input{ background-color:black;
78行目
              color:white;
79行目
              font-size: 1 em;
80行目
              }
81行目 input#a{ime-mode:avtive;
82行目
              }
83行目 input#i{ime-mode:inavtive;
84行目
              }
85行目
86行目 A{
              border:medium;
87行目
              border-style:solid;
88行目
              }
89行目 A:hover {color: black;
90行目
              background-color:white;
91行目
              }
92行目 A:forcus {color: black;
93行目
              background-color:white;
94行目
              }
95行目
96行目 a#ans{ color:black;
97行目
              background-color:black;
98行目
99行目 a#ans:hover{color:white;
```

```
100行目 background-color:black;
101行目 }
102行目
103行目 img {filter:invert()}
104行目 -->
```

Bw.css

```
1行目
       <!--
2行目
               font-size:1.5em;
       H1{
3行目
               margin-top:1.5em;
               margin-bottom:1.5em;
4行目
5行目
               text-align:center;
6行目
               line-height:1.3em;
7行目
               }
8行目
9行目
       H1:before{display:inline;
               counter-increment: "level1";
10行目
               content:counter("level1")"章 ";
11行目
               counter-reset: "level2";
12行目
13行目
               font-size:1.3em;
14行目 H2{
15行目
               margin-top:1em;
               margin-bottom: 1em;
16行目
17行目
               text-align:center;
               line-height:1.3em;
18行目
19行目
               }
20行目 H2:before{display: inline;
               counter-increment:"level2";
21行目
               content:"第"counter("level2")"節 ";
22行目
               content:"▼"
23行目
24行目
               }
       H1 + H2{margin-top:0em}
25行目
26行目
27行目 H3{
               font-size:1.2em;
28行目
               margin-top:1em;
29行目
               margin-bottom:0.5em;
30行目
               text-align:left;
```

```
line-height:1.3;
31行目
32行目
               margin-left:2em;
33行目 }
34行目 H2 + H3{margine-top:0em}
35行目
36行目 H4{
               font-size:1.1em;
37行目
               margin-top:1em;
38行目
               margin-bottom:0.5em;
39行目
               text-align:left;
40行目
               line-height: 1.3;
41行目
               margin-left:1.5em;
42行目 }
43行目
      H3 + H4{margine-top:0em}
44行目
45行目 H5{
               font-size:1em;
46行目
               margin-top:0.5em;
47行目
               margin-bottom:0.5em;
48行目
               text-align:left;
49行目
               line-height: 1.2;
50行目
               margin-left:1em;
51行目 }
      H4 + H5{margine-top:0em}
52行目
53行目
54行目 P{
               text-indent:1em}
55行目 p#idt01{margin-left:1em}
56行目 p#idt02{margin-left:2em}
57行目 p#idt03{margin-left:3em}
58行目 p#idt04{margin-left:4em}
59行目 p#idt05{margin-left:5em}
60行目 p#idt06{margin-left:6em}
61行目
62行目 山{
               margin-left:1em;
63行目
               }
64行目
      table{ font-size:1em;
65行目
               border-style:outset;
66行目
               margin-top:1.5em;
67行目
               margin-bottom: 1.5em;
68行目
69行目 th,td{
               border-style:outset;
```

```
70行目
             }
71行目
72行目 sup,sub{font-size:90%;
73行日
              }
74行目
75行目 input{ background-color:white;
76行目
              color:black;
77行目
              font-size: 1 em;
78行目
              ime-mode:active;
79行目 input#a{ime-mode:avtive;
80行目
              }
81行目 input#i{ime-mode:inavtive;
82行目
              }
83行目
84行月 A{
             border:medium;
85行目
              border-style:solid;
86行目
              }
87行目 A:hover {color:white;
88行目
              background-color:black;
89行目
              }
90行目
91行目 a#ans{ color:white;
92行目
              background-color:white;
93行目
94行目 a#ans:hover{color:black;
95行目
              background-color:white;
96行目
              }
97行目 -->
```

各被験者の読字数

		課題											
ID	実施日	番号	メディア	1回目	2回目	3回目	4回目	5回目	6回目	7回目	8回目	9回目	
	1999/10/5	1	нс	352.94	318.06	280.94	432.73	381.79	359.93	359.06	424.78	272.93	
AA .	1999/10/5	2	нтм	439.69	354.15	309.85	380.80	291.44	318.61	352.40	289.25	384.82	
	1999/10/5	3	PDF	379.83	404,72	383.53	290.48	216.77	313.99	305.08	272.29	354.00	
АВ	1999/9/29	2	HC	383.70	329.25	340.06	470.67	324.73	368.39	391.08	284.28	405.18	
	1999/9/29	3	нтм	326.06	248.49	321.82	306,31	265.64	306.37	323.56	295.30	377,56	
	1999/9/29	1	PDF	364.12	356.61	293.29	397.99	274.21	398.68	350.23	443.72	354.52	
AD .	1999/9/28	2	HC	450.20	328.37	354.89	433.25	428.85	412.53	438.18	373.55	434.26	
	1999/10/28	3	нтм	422.96	459.48	428.05	380.39	407.99	457.36	330.25	370.49	401.56	
	1999/9/28	1	PDF	378.56	379.14	344.36	541.73	430.55	331.33	387.76	473.46	404.49	
AE .	1999/9/28	1	HC	252.33	268,11	212.42	274.95	267.16	245.54	282.29	302.41	266.27	
	1999/9/28	2	нтм	381.11	253.73	296.86	314.12	238.77	246.44	296.10	284.52	284.57	
	1999/9/28	3	PDF	293.89	275.65	284.11	203.01	268.80	308.55	213.27	218.71	292.09	
AG	1999/9/29	3	HC	313.02	320.37	369.71	249.56	255.07	288.71	223.38	224.06	274.60	
	1999/9/29	1	нтм	246.28	320.54	261.06	330.71	323.92	326.33	296.36	382.19	283.24	
	1999/9/29	2	PDF	468.88	407.65	383.53	430.12	295.39	268.14	312.04	330.87	288.92	
	1999/9/30	2	HC	429.93	369.40	412.11	491.40	448.25	382.58	413.29	417.23	401.11	
AH	1999/9/30	3	нтм	400.71	392.88	408.78	392.09	363.74	363.09	373.16	339.51	353.29	
•	1999/9/30	1	PDF	435.17	443.65	315.09	504.95	405.20	476.42	407.87	466.62	371.90	
	1999/9/26	2	HC	486.02	327.85	308.36	447.93	369.70	365.28	340.81	334.48	402.23	
AI	1999/9/26	1	нтм	237.89	276.02	279.29	364.29	333.67	355.63	253.52	290.49	317.32	
•	1999/9/26	3	PDF	320.72	355.99	325.67	345.76	335.73	357.52	320.83	265.57	309.25	
	1999/10/3	1	нс	298.55	190.68	209.10	311.79	228.43	288.85	207.85	257.99	209.79	
BA	1999/10/3		нтм	182.41	190,74	185.68	218.55	148.30	167.50	226.64	195.11	190.78	
	1999/10/3		PDF	測定不能									
	1999/9/30		HC	219.56	214.14	183.75	272.42	214.78	262.34	201.06	276,40	237.47	
вс	1999/9/30		нтм	172.43	154.00	170.11	162.90	168,08	142.42	142.62	147.52	177.69	
	1999/9/30		PDF	測定不能									
	1999/10/1	3	HC	162.01	159.83	148.83	115.70	186.86	158.39	163.83	160.66	164.72	
BD .	1999/10/1	i	нтм	101.33	117.71	93.64	141.61	128.69	101.64	104.54	148.29	112.69	
•	1999/10/1	2	PDF	測定不能									
	1999/9/26		HC	249.07	225.96	175.28	282.15	245.77	270.15	234.89	278.44	254,69	
BE .	1999/9/26		нтм	261.88	241.88	283.65	193.50	219.83	224.40	217.75	228.90	268.76	
•	1999/9/26		PDF	291.79	253.31	222.36	296.61	263.49	264.63	281.21	239.57	226.31	
	1999/10/3		нс	232.27	192.10	202.11	220.64	242.31	266.26	172.60	261.27	223.67	
cc	1999/10/3		нтм	232.45	247.79	213.76	210.19	198.63	274.58	230.46	152.26	216.89	
	1999/10/3		PDF	258.38	209.00	197.16	262.22	214.50	187.15	231.00	191.09	222.57	
	1999/10/6		HC	313.87	263.20	274.95	371.68	347.27	326.33	316.96	380.47	333.18	
CD	1999/10/6		нтм	295.42	259.12	263.00	358.61	262.55	299.70	329.00	283.90	307.96	
	1999/10/6		PDF	242.14	295.37	279.84	260.01	289.00	292.19	300.13	250.46	245.88	
	1999/10/8		HC	579.49	459.82	423.87	662.34	492.47	478.30	471,40	449.30	580.18	
CE	1999/10/8		нтм	483.94	495.99	469.50	448.77	400.24	415.35	378,53	351.66	425.28	
-	1999/10/8		PDF	517.24	416.16	344.66	604.06	461.54	488.32	442.15	578.38	488.80	
	1999/10/7		HC	316.23	339.37	356.14	310.98	335.21	399.74	362.86	338.49	372.29	
CF	1999/10/7		нтм	481.53	361.83	417.21	461.84	389.29	399.47	387.76	341.91	327.87	
	1999/10/7		PDF	400.95	368.74	285.48	489.04	423.03	449.55	372.96	433.50	343.51	
CG	1999/10/7		HC	191.74	171.67	163.59	125.57	146.28	174,39	159.70	158.38	148.03	
			HTM	128.02	130.32	126.56	221.12	141.22	185.04	148.18	174.20	131.35	
	1999/10/7		PDF	138.62	142.33	112.08	172.13	174.15	132.84	171.34	150.20	124.27	
CI	1999/10/3		HC	334.81	331.90	343.69	273.56	317.48	291.77	303.60	314.91	277.03	
	1999/10/3		нтм	257.79	241.12		348.29	317.01	308.08	305.49	316.67	281.69	
	1999/10/3		PDF	344.16	287.98	335.23	304.87	327.16	203.83	375.82	282.03	303.41	
	1999/9/28		HC.	163.10		151.12		159.82	282.78	130.46	155.08	184.28	
			нтм	171.21	132.33	156.53		175.71	185.26	160.30	174,43	151.74	
	1999/9/28		PDF	66.24	91.41	57.99		122.15	90.24	135.39	124.43	111.65	
	1333/3/20		FUF	00.24	31.41	31.38	122.73	144.10	30.24	. 50.03	, 24,40		

		課題		検索時間(sec)								
ID	実施日	番号	メディア	1回目	2回目	3回目	4回目	5回目	6回目	7回目	8回目	9回目
	1999/10/5		ЦС	10.95	8.1	14.52	17.93	11.75	10.65	10.9	9.91	18.53
AA	1999/10/5	2	HC HTM	2.66	11.2	8.52	20.11	15.29	6.46	11.64	10.9	4.3
	1999/10/5	3	PDF	13.57	10.21	7.82	7.86	20.44	12.33	11.64	14.84	16.03
		2			22.01	15.14		9.6	17.43	12.9	16.71	13.02
АВ	1999/9/29	3	HC HTM	5.33	13.13	14.28	11.32 4.36	11.82	11.85	7.21	13.49	10.32
	1999/9/29	-3 -	PDF	9.09 15.65	5.57	19.48	11.54	8.63	8.35	14.48	6	12.81
AD	1999/9/28	2	HC	4.53	8.23	12.05	9.62	11.07	19.82	12.28	13.19	7.85
	1999/10/28	3	нтм	4.64	16.3	16.95	13.85	21.27	7.53	21.35	16.86	10.81
7.0	1999/9/28	1	PDF	24.27	14.06	29	25.05	14	10.18	9.35	6.15	15.29
	1999/9/28	<u> </u>	HC	18.09	13.31	24.09	21.12	19.24	12.36	15.29	10.08	16.71
ΑE	1999/9/28	2	нтм	2.6	13.26	11.17	10.06	17.07	8.53	15.02	10.49	5.57
	1999/9/28	3	PDF	17.81	13.81	11.83	7.42	22.97	23.46	8.05	13.82	20.14
	1999/9/29	3	HC	14.93	21.82	12.16	9.02	21.64	24.16	8.09	14.42	20.88
AG	1999/9/29	1	НТМ	6.52	5.25	8.48	7.1	7.49	18.03	6.75	7.23	8.01
	1999/9/29	2	PDF	1.85	7.85	8.67	8.18	16.78	6.6	10.47	7.92	7.96
	1999/9/30	2	НС	6.31	8.3	9.21	9.45	15.57	10.89	9.71	8.4	8.98
АН	1999/9/30	3	нтм	12.25	8.11	8.45	7.32	30.77	8.84	4.93	18.37	7.84
	1999/9/30	1	PDF	6.33	25.7	8.42	27.25	9.17	8.13	7.89	6.75	13.82
	1999/9/26	2	HC	4.73	7.36	11.46	18.66	17.64	16.38	29.5	12.64	10.21
ΑI	1999/9/26	1	нтм	5.78	10.74	10.22	9.54	20.21	13.86	16.64	13.27	24.38
	1999/9/26	3	PDF	19.34	34.83	23.27	8.33	30.88	14.21	22.21	14.27	13.65
	1999/10/3	1	HC	31.01	18.87	33.42	34.03	23.35	36.42	25.11	20.6	29.89
BA	1999/10/3	2	нтм	11.84	25.92	31.8	28.99	52.22	6.14	21.38	31.03	8.64
	1999/10/3	3	PDF	測定不能								
	1999/9/30	1	HC	29.56	27.41	43.25	53.09	29.23	26.47	86.28	20.4	53.33
BC	1999/9/30	3	нтм	61.4	45.12	53.68	29.41	63.5	46.13	26.74	50.79	46.48
	1999/9/30	2	PDF	測定不能								
	1999/10/1	3	HC	29.67	35.35	72.77	21.76	50.46	37.25	58.1	33.96	44.4
BD	1999/10/1	1	нтм	37.73	40.74	84.88	65.88	41.99	106.57	38.27	32.58	82.7
	1999/10/1	2	PDF	測定不能								
	1999/9/26	1_	HC	33.27	76.5	24.64	27.25	19.27	10.45	13.74	19.32	26.18
BE	1999/9/26	3	НТМ	30.83	23.57	14.11	22.35	40.86	22.46	16.64	41.18	34.87
	1999/9/26	2	PDF	39	17.37	31.46	27.17	71.28	32.36	32.21	46.21	11.14
	1999/10/3	1	нс	15.24	15.13	26.29	29.33	32.46	13.27	14.51	14.67	17.71
CC	1999/10/3	3	НТМ	7.43	6.35	7.85	8.56	8.72	10.31	4.93	9.72	11.64
	1999/10/3	2	PDF	7.13	16.41	13.31	38.46	20.91	8.18	19.64	13.14	7.79
	1999/10/6		HC	7.84	23.21	18.91	11.36	18.23	7.62	25.11	7.14	15.08
CD	1999/10/6		НТМ	5.59	22.05	16.06	13.26	10.44	5.44	17.03	14.77	7.57
	1999/10/6		PDF	12.34	9.93	11.2	5.56	31.89	31.49	12.04	11.82	22.01
	1999/10/8		HC	8.14	11.67	19.25	10.14	34.6	8.7	10.72	25.07	8.21
CE	1999/10/8		HTM	8.44	8.41	7.89	5.51	13.64	9.14	5.96	11.82	11.09
	1999/10/8		PDF	14.28	10.02	16.43	15.64	9.31	14.07	9.42	11.44	21.69
	1999/10/7		HC	13.92	10.57	7.68	7.46	7	10.33	14.93	7.53	10.53
CF	1999/10/7		HTM	2.68	19.07	5.17	9.85	11.25	17.93	11.06	11.44	7.71
	1999/10/7		PDF	3.58	10.91	7.25	7.24	4.09	6.21	7.37	14.93	14.29
CG	1999/10/7		HC	33.07	39.69	23.71 30.25	20.99	30.86 11.57	31.46 12.75	39.88 11.43	26.5 10.92	26.64 26.71
ÇĞ	1999/10/7		HTM PDF	14.88 10.35	8.07 33.47	42.44	61.31	43.96	16.4	39.17	43.02	90.32
	1999/10/7 1999/10/3		HC	21.21	13.07	16.74	11.13	19.91	13.93	8.17	13.09	12.36
CI	1999/10/3		НТМ	10.31	13.16	17.21	16.19	10.98	23.8	10.56	5.89	17.35
01	1999/10/3		PDF	8.24	12.04	23.08	18.01	28.57	9.68	19.61	39.1	11.38
CJ	1999/9/28		HC	43.89	47.12	29.15	12.17	35.5	22.85	21.43	23.4	19.62
	1999/9/28		HTM	25.16	4.64	24.82	16.23	10.25	10.82	12.21	10.97	48.33
	1999/9/28		PDF	28.24	21.46	21.58	55.19	24.07	15.99	60.48	29.25	11.49
					•							

3 読 材料

実験で用いた読材料

実験用課題

- ・はなの みち(p.1)
- ・おはなし きいて(p.2)
- ・だれに あえるかな(p.3)
- ・とりと なかよし(p.4)
- ・じどう車くらべ(p.5)
- ・ものの 名まえ(p.6)
- ・すうっと、すっと、大すきた よ(p.7)
- ・どうぶつの 赤ちゃん(p.8)
- ・きりかぶの 赤ちゃん(p.9)
- ・くじらぐも(p.10)
- ・ふきのとう(p.11)
- ・たんぱぽの ちえ(p.12)
- ・王さま 出かけましょう
- (p.13)
- ・えいっ(p.14)
- ・かけふみおにんをしよう
- (p.15)
- ・お手紙 (p.16)

- ・こんな話を考えた(p.17)
- ・おへそってなあに(p.18)
- ・こまを回せるようになったよ (p.19)
- ・スイミー(p.20)
- ・森に生きる(p.21)
- ・ヤモリをつかまえた(p.22)
- ・ヤドカリのすみかえ(p.23)
- ・ありの行列(p.24)
- ・エルマー、トラに会う
- (p.25)
- ・三年とうげ (p. 26)
- ・たこたこあがれ(p.27)
- ・ちいちゃんのかげおくり
- (p.2B)
- ・虫のゆりかご(p.29)
- ・つり橋わたれ(p.30)

はなの みち くまさんが、ふくろを みつ けました。

「おや、なにかな。いっぽい はいって いる。」

くまさんが、ともだちの りすさん に、ききに いきました。

くまさんが、ふくろを あけました。 なにも ありません。

「しまった。あなが あいて い た。」

あたたかい かぜが ふきはじめま した。

次は、「エルマー、とらに会う」です。

がっこう、てつぼう、のぼり ぼう、げんきに、たいそう、た のしいな。

次は、「虫のゆりかご」です。

1

2

たれに あえるかな うさぎの ひょんは、はやお きです。「きょうは、だれに あえるかな。たのしみ、たのし み。」

ひょんは、さんぱに でかけ ました。

なんて いい てんきでしょう。「おひさま、おはよう。」 ひょんが、ひょんひょん とんで いくと、たんぽぽが さいて いました。

次は、「こまを回せるように なったよ」です。 とりと なかよし これは、わにちどりです。 わにちどりは、かわの そば に すんで います。 わにちどりと なかよしの

わにちとりと なかよしの どうぶつは、なんでしょう。 それは、わにです。

わにの はには、たべもののかすが、たくさん ついて います。

わにちどりは、かすを たべて、わにの はを きれいに するのです。

次は、「三年とうげ」です。

3

じどう車くらべ いろいろな じどう車が、どうろを はしって います。

それぞれの じどう車は、どんな しごとを して いますか。

その ために、どんな つくりに なって いますか。

バスや じょうよう<mark>単は、人を</mark>のせて はこぶ しごとを して います。

その ために、ざせきの ところが、 ひろく つくって あります。

以上です。

ものの 名まえ

けんじくんは、夕がた、おねえさんと かいものに いきました。はじめの おみせには、りんご、みかん、バナナなどが、ならんで います。ふたりは、五百円でりんごを かいました。 この おみせは、なにやさんでしょう。

つぎに、さかなやさんにいきました。

次は、「たこたこあがれ」です。

5

6

ずうっと、ずっと、大すきだよ エルフの ことを はなします。 エルフは、せかいで いちばん す ばらしい 犬です。

ぼくたちは、いっしょに 大きくなった。でも、エルフのほうが、ずっと 早く、大きく なったよ。

ばくは、エルフの あったかい おなかを、いつも まくらに するのが すぎだった。

次は、「たんぽぽの ちえ」です。

どうぶつの 赤ちゃん

どうぶつの 赤ちゃんは、生まれたばかりの ときは、どんな ようすをして いるのでしょう。そして、どのように して 大きく なって いくのでしょう。

ライオンの 赤ちゃんは、生まれた ときは、子ねこぐらいの 大きさです。 目や 耳は、とじた ままです。

次は、「ふきのとう」です。

7

きりかぶの 赤ちゃん どうぶつの村の 入り口に、大きな

どうぶつの村の 入り口に、大きなきりかぶが 一つ ありました。このきりかぶは、きられる まえには、天まで とどく 大きな 大きな 木だったのです。

でも、きょ年の ふゆに、どうぶつ たちが それを きって、たきぎに して しまったのです。

次は、「王さま 出かけましょう」です。

くじらぐも

四じかんめの ことです。

ーねん二くみの 子どもたちが たいそうを して いると、 空に、大きな くじらが あらわれました。まっしろい くもの くじらです。

「一、二、三、四。」 くじらも、たいそうを はじめまし た。のびたり ちぢんだり して、し

次は、「つり橋わたれ」です。

んこぎゅうも しました。

9

1 0

ふきのとう よが あけました。 朝の 光を あびて、 竹やぶの 竹の はっぱが、 「さむかったね。」 「うん、さむかったね。」 と、ささやいて います。 曾か まだ すこし のこって、 あたりは しんと して います。

た。 「よいしょ、よいしょ。おもたい な。」

竹やふの そばの ふきのとうです。

どこかで、小さな こえが しまし

次は、「ものの 名まえ」です。

たんぱぱの ちえ 春に なると、たんぱぱの質色い きれいな 花がさぎます。

二、三日 たつと、その花は しばんで、だんだん黒っぱい 色に かわっていきます。そうして、たんぱぱの花の じくは、ぐったりと 地面にたおれて しまいます。

けれども、たんぱぱは、かれてしまったのでは ありません。

次は、「森に生きる」です。

1 1

王さま 出かけましょう 王さま へやの まどの 上に、こ としも つばめが やって きました。 つばめは、いっしょうけんめい すを 作りなおしています。

王さまは、気に なります。ときどき、そっと のぞいたり します。 「もう できたかな。」 と、べんきょうの 時間に なって も、よそ見ばかり して います。

次は、「おはなし きいて」です。

えいっ

くまの とうさんと くまの 子ど もが、町へ ポップコーンを 買いに 行きました。

歩いて いると、どうろの しんごうが、赤に なりました。「あぶないから、わたっては いけないよ。」と、とうさんが 含いました。「うん。」

くまの 子が 含いました。 車が、どんどん 通って いきます。

次は、「どうぶつの 赤ちゃん」で す。

1 3

14

かげふみおにをしよう

かげふみおにができるのは、どんな 天気の日でしょう。

雨ふりの日には、かげはできません。くもった日も、かげはうすくで、はっきりしません。晴れた日でも、太陽が雲にかくれて、日がささなくなると、かげはできません。

次は、「とりと なかよし」です。

お手紙

がまくんは、げんかんの前に すわっていました。

かえるくんがやってきて、食いました。「どうしたんだい、がまがえるくん。きみ、かなしそうだね。」

「うん、そうなんだ。」 がまくんが 含いました。

「今、一日のうちの かなしい時なんだ。つまり、お手紙をまつ時間なんだ。」

次は、「はなの みち」です。

1 5

こんな話を考えた

つぎのページの絵を 見て、お話を 作りましょう。

まず、絵を見くらべます。そして、 どんなお話にするかを考えながら、ど の絵と どの絵を選び、どんなじゅん じょにならべるかを きめましょう。 つぎに、どんなだいがいか、どの場 面を くわしく書くかを、考えましょう。

以上です。

おへそってなあに

「おへそは、どこにあるの。」ときかれたら、「知っているよ、そんなこと。おなかのまん中。」と、だれでも含うでしょう。でも、「おへそは、なんのためにあるの。」ときかれたら、あなたは答えられますか。

わたしたちのおへそは、今は、なん のはたらきもしていません。

·次は、「ちいちゃんのかげおくり」です。

1 7

18

こまを回せるようになったよこの前、生活化の時間に、ふれ合い学しゅうがありました。ふれあい学しゅうの先生は、近くにすんでいるおじいさんやおばあさんです。

お手玉や、こま回しなどのあそびを 教わりました。その中で、ぼくがいち ばん楽しかったのは、こま回しです。

次は、「ヤドカリのすみかえ」です。

スイミー

広い 海の どこかに、小さな 魚の 兄弟たちが、楽しく くらしていた。

みんな 赤いのに、一びきだけは、 からす貝よりもまっ黒。およぐのは、 だれよりも はやかった。

名まえは、スイミー。

あるひ、おそろしい まぐろが、おなかを すかせて、すごい はやさで ミサイルみたいに つっこんで きた。

次は、「くじらぐも」です。

19

森に生きる

真ん丸な顔に光る、くりくりした二つの目。フクロウは、ひと目見ても、 やあ、と声をかけたくなる鳥だ。

けれど、昼間はひっそりとくらしている。木のこぶみたいにしか見えないから、なかなか出会えない。そんなフクロウが、夜になると、すばらしいかりゅうどになる。

次は、「おへそってなあに」です。

ヤモリをつかまえた

「キャー。」

お母さんのひめいが、二階から聞こ えてきました。

わたしとお姉さんは、急いで声のする方へかけていきました。すると、二階の階だんの小まどの所で、ねずみ色のヤモリが一びき、あみ戸を上っているところでした。

次は、「えいっ」です。

2 1

2 2

ヤドカリのすみかえ

ヤドカリは、まき貝のからの中にすんでいます。貝がらは大きくなりません。それで、ヤドカリは、体の成長に合わせて、なんどかすみかえをします。そのとき、空の貝がらに入ることもありますが、ほかのヤドカリの貝がらにすみかえることが多いのです。

次は、「じどう軍くらべ」です。

ありの行列

夏になると、庭のすみなどで、ありの行列をよく見かけます。その行列は、ありの巣から、えさのある所まで、ずっとつづいています。ありは、ものがよく見えません。それなのに、なぜ、ありの行列ができるのでしょうか。

アメリカに、ウイルソンという学者がいます。

次は、「きりかぶの 赤ちゃん」で す。 ェルマー、とらに会う

これは、「エルマーのぼうけん」と いう本に入っているお話です。

エルマーは、年とったねこから、動物島にとらえられているりゅうの子の話を聞きました。そして、とてもかわいそうに思いました。

一度でいいから、りゅうのせなかに のって空をとんでみたいと思っていた エルマーは、このりゅうの子をたすけ に行くことにしました。

次は、「ヤモリをつかまえた」です。

三年とうげ

あるところに、三年とうげとよばれ るとうげがありました。

あまり高くない、なだらかなとうげ でした。

番には、すみれ、タンポポ、ふでりんどう。とうげからふもとまでさきみだれました。れんげつつじのさくころは、だれだってため息の出るほど、よいながめでした。

次は、「こんな話を考えた」です。

25

26

たこたこあがれ

たこあげは、昔から子どもの遊びと して、また、地いきの行事として楽し まれてきました。今も、日本各地には、 その土地どくとのたこがあるでしょうか。 また、それらのたこは、どんなねがい をこめて作られ、大空にあげられるの でしょうか。

以上です。

ちいちゃんのかげおくり

「かげおくり」って遊びをちいちゃんに教えてくれたのは、お父さんでした。

出征する前の日、お父さんは、ちいちゃん、お兄ちゃん、お母さんをつれて、先祖のはかまいりに行きました。 その帰り道、青い空を見上げたお父さんが、つぶやきました。

次は、「だれに あえるかな」です。

虫のゆりかご

ここは、雑木林の中。あちらこちらに、農家が見える。

おやっ、おもしろい物を見つけた。 くるくるときれいにまかれた葉っぱが、 地面に落ちている。あっちにもある。 こっちにもある。

その一つを手に取って、そっと広げ てみる。一まいの葉っぱになった。

次は、「かげふみおにをしよう」で す。 つり橋わたれ

「やあい、やあい、くやしかったら、つり橋わたって、かけてこい。」 山の子どもたちがはやしました。 トッコは、きゅっとくちびるをかみ しめて、ゆれるつり橋を見ました。ふ じづるでできた橋の下には、谷川がゴ ーゴーとしぶきを上げて流れています。

以上です。

29

