

MediaWiki:Lékařská kalkulačka/NIHSS.js

Note: After publishing, you may have to bypass your browser's cache to see the changes.

- **Firefox / Safari:** Hold *Shift* while clicking *Reload*, or press either *Ctrl-F5* or *Ctrl-R* (*⌘-R* on a Mac)
- **Google Chrome:** Press *Ctrl-Shift-R* (*⌘-Shift-R* on a Mac)
- **Internet Explorer / Edge:** Hold *Ctrl* while clicking *Refresh*, or press *Ctrl-F5*
- **Opera:** Press *Ctrl-F5*.

```
1  /**
2   * ReplaceAll by Fagner Brack (MIT Licensed)
3   * Replaces all occurrences of a substring in a string
4   */
5  String.prototype.replaceAll = function(token, newToken, ignoreCase) {
6      var str, i = -1, token;
7      if((str = this.toString()) && typeof token === "string") {
8          token = ignoreCase === true? token.toLowerCase() : undefined;
9          while((i =
10             token !== undefined?
11             str.toLowerCase().indexOf(
12                 token,
13                 i >= 0? i + newToken.length : 0
14             ) : str.indexOf(
15                 token,
16                 i >= 0? i + newToken.length : 0
17             )
18             ) !== -1 ) {
19              str = str.substring(0, i)
20                  .concat(newToken)
21                  .concat(str.substring(i + token.length));
22          }
23      }
24      return str;
25  };
26
27  /***** Převod do html podoby *****/
28  var content = $( '#lekarska_kalkulacka_NIHSS-content' ).text();
29  $( '#lekarska_kalkulacka_NIHSS' ).html( content.replaceAll('paragraph', 'p').replaceAll('resValue', 'span')
30  ).replaceAll('breakline', 'br').replaceAll('division', 'div').replaceAll('orderedlist', 'ol' );
31
32
33  /***** Vlastní výpočet *****/
34  $( 'input' ).change( function() {
35      var NIHSS1a = parseInt($('input[name=NIHSS1a]:checked').val());
36      var NIHSS1b = parseInt($('input[name=NIHSS1b]:checked').val());
37      var NIHSS1c = parseInt($('input[name=NIHSS1c]:checked').val());
38      var NIHSS3 = parseInt($('input[name=NIHSS3]:checked').val());
39
40      //promenne, kde vyznam v kodu nese desetinne misto
41      var NIHSS2 = parseFloat($('input[name=NIHSS2]:checked').val());
42      var NIHSS4 = parseFloat($('input[name=NIHSS4]:checked').val());
43      var NIHSS8 = parseFloat($('input[name=NIHSS8]:checked').val());
44      var NIHSS9 = parseFloat($('input[name=NIHSS9]:checked').val());
45      var NIHSS11 = parseFloat($('input[name=NIHSS11]:checked').val());
46
47      //promenne, kde je soucasti moznosti undefined - "UN"
48      var NIHSS5dx = $('input[name=NIHSS5dx]:checked').val();
49      var NIHSS5sin = $('input[name=NIHSS5sin]:checked').val();
50      var NIHSS6dx = $('input[name=NIHSS6dx]:checked').val();
51      var NIHSS6sin = $('input[name=NIHSS6sin]:checked').val();
52      var NIHSS7 = $('input[name=NIHSS7]:checked').val();
53      var NIHSS10 = $('input[name=NIHSS10]:checked').val();
54
55      var nihss = 0;
56      nihss += NIHSS1a + NIHSS1b + NIHSS1c + Math.round(NIHSS2) + NIHSS3 + Math.round(NIHSS4) + Math.round(NIHSS8) + Math.round(NIHSS9) +
57      Math.round(NIHSS11);
58      if (NIHSS5dx != 'UN') { nihss += parseInt(NIHSS5dx); }
59      if (NIHSS5sin != 'UN') { nihss += parseInt(NIHSS5sin); }
60      if (NIHSS6dx != 'UN') { nihss += parseInt(NIHSS6dx); }
61      if (NIHSS6sin != 'UN') { nihss += parseInt(NIHSS6sin); }
62      if (NIHSS7 != 'UN') { nihss += parseInt(NIHSS7); NIHSS7 = parseInt( NIHSS7 ); }
63      if (NIHSS10 != 'UN') { nihss += parseInt(NIHSS10); NIHSS10 = parseFloat(NIHSS10); }
64
65      var result = "<b>NIHSS:</b> " + nihss + ".\n<p>";
66
67      if (NIHSS1a > 0) {
68          switch (NIHSS1a) {
69              case 1: result += "Somnolence &ndash; 1. "; break;
70              case 2: result += "Sopor &ndash; 2. "; break;
71              case 3: result += "Kóma &ndash; 3. "; break;
72          }
73      }
74      if (NIHSS1b > 0) {
75          switch (NIHSS1b) {
76              case 1: result += "Slovní odpovědi &ndash; 1. "; break;
77              case 2: result += "Slovní odpovědi &ndash; 2. "; break;
78          }
79      }
80      if (NIHSS1c > 0) {
81          switch (NIHSS1c) {
82              case 1: result += "Vyhovění výzvam &ndash; 1. "; break;
83              case 2: result += "Vyhovění výzvam &ndash; 2. "; break;
84          }
85      }
86      if (NIHSS2 > 0) {
87          switch (NIHSS2) {
88              case 1.0: result += "Okohybná paréza &ndash; 1. "; break;
```

```

89         case 1.1: result += "Potlačitelná pohledová paréza &ndash; 1. "; break;
90         case 2: result += "Nepotlačitelná pohledová paréza &ndash; 2. "; break;
91     }
92 }
93 if (NIHSS3 > 0) {
94     switch (NIHSS3) {
95         case 1: result += "Částečná hemianopsie &ndash; 1. "; break;
96         case 2: result += "Kompletní hemianopsie &ndash; 2. "; break;
97         case 3: result += "Slepota &ndash; 3. "; break;
98     }
99 }
100 if (NIHSS4 > 0) {
101     switch (NIHSS4) {
102         case 1: result += "Frustrní faciální paréza dolní větve &ndash; 1. "; break;
103         case 2.0: result += "Částečná faciální paréza dolní větve &ndash; 2. "; break;
104         case 2.1: result += "Kompletní faciální paréza dolní větve &ndash; 2. "; break;
105         case 3.0: result += "Faciální paréza horní i dolní větve &ndash; 3. "; break;
106         case 3.1: result += "Bilaterální faciální paréza &ndash; 3. "; break;
107         case 3.2: result += "Kóma &ndash; 3. "; break;
108     }
109 }
110 if (NIHSS5dx != 'UN' && NIHSS6dx != 'UN' && NIHSS5sin != 'UN' && NIHSS6sin != 'UN' && NIHSS5dx > 0 && NIHSS6dx > 0 && NIHSS5sin > 0
&& NIHSS6sin > 0) {
111     result += "Kvadruparéza &ndash; " + parseInt(NIHSS5dx) + "+" + parseInt(NIHSS5sin) + "+" + parseInt(NIHSS6dx) + "+" +
parseInt(NIHSS6sin) + ". ";
112 } else if (NIHSS5dx != 'UN' && NIHSS6dx != 'UN' && NIHSS5dx > 0 && NIHSS6dx > 0) {
113     result += "Pravostranná hemiparéza &ndash; " + parseInt(NIHSS5dx) + "+" + parseInt(NIHSS6dx) + ". ";
114     if (NIHSS5sin != 'UN' && NIHSS5sin > 0){
115         result += "Paréza levé horní končetiny &ndash; " + parseInt(NIHSS5sin) + ". ";
116     }
117     if (NIHSS6sin != 'UN' && NIHSS6sin > 0){
118         result += "Paréza levé dolní končetiny &ndash; " + parseInt(NIHSS6sin) + ". ";
119     }
120 } else if (NIHSS5sin != 'UN' && NIHSS6sin != 'UN' && NIHSS5sin > 0 && NIHSS6sin > 0) {
121     result += "Levostranná hemiparéza &ndash; " + parseInt(NIHSS5sin) + "+" + parseInt(NIHSS6sin) + ". ";
122     if (NIHSS5dx != 'UN' && NIHSS5dx > 0){
123         result += "Paréza pravé horní končetiny &ndash; " + parseInt(NIHSS5dx) + ". ";
124     }
125     if (NIHSS6dx != 'UN' && NIHSS6dx > 0){
126         result += "Paréza pravé dolní končetiny &ndash; " + parseInt(NIHSS6dx) + ". ";
127     }
128 } else if (NIHSS6dx != 'UN' && NIHSS6sin != 'UN' && NIHSS6dx > 0 && NIHSS6sin > 0) {
129     result += "Paraparéza &ndash; " + parseInt(NIHSS6dx) + "+" + parseInt(NIHSS6sin) + ". ";
130 } else {
131     if (NIHSS5dx != 'UN' && NIHSS5dx > 0){
132         result += "Paréza pravé horní končetiny &ndash; " + parseInt(NIHSS5dx) + ". ";
133     }
134     if (NIHSS5sin != 'UN' && NIHSS5sin > 0){
135         result += "Paréza levé horní končetiny &ndash; " + parseInt(NIHSS5sin) + ". ";
136     }
137     if (NIHSS6dx != 'UN' && NIHSS6dx > 0){
138         result += "Paréza pravé dolní končetiny &ndash; " + parseInt(NIHSS6dx) + ". ";
139     }
140     if (NIHSS6sin != 'UN' && NIHSS6sin > 0){
141         result += "Paréza levé dolní končetiny &ndash; " + parseInt(NIHSS6sin) + ". ";
142     }
143 }
144 }
145 if (NIHSS7 != 'UN' && NIHSS7 > 0) {
146     switch (NIHSS7) {
147         case 1: result += "Ataxie &ndash; 1. "; break;
148         case 2: result += "Ataxie &ndash; 2. "; break;
149     }
150 }
151 }
152 if (NIHSS8 > 0) {
153     switch (NIHSS8) {
154         case 1: result += "Lehká porucha čítí &ndash; 1. "; break;
155         case 2.0: result += "Těžká porucha čítí unilaterální &ndash; 2. "; break;
156         case 2.1: result += "Těžká porucha čítí bilaterální &ndash; 2. "; break;
157         case 2.2: result += "Kóma &ndash; 2. "; break;
158     }
159 }
160 if (NIHSS9 > 0) {
161     switch (NIHSS9) {
162         case 1.0: result += "Lehká expresivní fatická porucha &ndash; 1. "; break;
163         case 1.1: result += "Lehká senzorická fatická porucha &ndash; 1. "; break;
164         case 2.0: result += "Těžká expresivní fatická porucha &ndash; 2. "; break;
165         case 2.1: result += "Těžká senzorická fatická porucha &ndash; 2. "; break;
166         case 3.0: result += "Globální afázie &ndash; 3. "; break;
167         case 3.1: result += "Kóma &ndash; 3. "; break;
168     }
169 }
170 if (NIHSS10 != 'UN' && NIHSS10 > 0) {
171     switch (NIHSS10) {
172         case 1: result += "Lehká dysartrie &ndash; 1. "; break;
173         case 2.0: result += "Těžká dysartrie &ndash; 2. "; break;
174         case 2.1: result += "Mutismus &ndash; 2. "; break;
175         case 2.2: result += "Kóma &ndash; 2. "; break;
176     }
177 }
178 if (NIHSS11 > 0) {
179     switch (NIHSS11) {
180         case 1: result += "Neglect &ndash; 1. "; break;
181         case 2.0: result += "Neglect &ndash; 2. "; break;
182         case 2.1: result += "Kóma &ndash; 2. "; break;
183     }
184 }
185 }
186
187 result += "</p>";
188 $( '#lekarska_kalkulacka_NIHSS-vysledek' ).html( result );
189 });

```