# Interpolacja wielomianowa

Kamil Górny, Wojciech Jachimowski, Mateusz Jendernal, Dominika Glugla

## Spis treści

1	$\mathbf{W}\mathbf{s}^{\dagger}$	tęp       3         Opis implementacji algorytmu realizującego metodę	
2	Dzi	ałanie programu 4	Ŀ
3	Opi	s programu 5	,
	$3.\bar{1}$	Struktura programu	,
	3.2	Wejście-wyjście	,
		Kod programu	;
		3.3.1 JavaScript	;
		3.3.2 HTML	)
		3.3.3 CSS - fragment	

#### 1 $\mathbf{Wstep}$

Treść zadania: "Program, który przewidzi rozwój przestępczości na świecie. Węzły mają przedstawiać liczbę popełnianych przestępstw na świecie zmieniającą się w czasie."

Wykorzystywana metoda numeryczna: metoda Hermite'a (postać Newtona z wykorzystaniem węzłów dwukrotnych)

Dla danych  $x_0, x_1, x_2, ...x_n$  parami różnych, dla dodatnich liczb naturalnych  $m_0,m_1,m_2,...m_n$  i dla znanych wartości  $f^{(i)}(x_j),j=0,1,...,n,i=0,1,...,m_j-1$ , szukamy wielomianu  $P\in \sqcap_m$ , gdzie  $m+1=\sum_{j=0}^n m_j$ , takiego, że

$$P^{(i)}(x_j) = f^{(i)}(x_j), j = 0, 1, ..., n, i = 0, 1, ..., m_j - 1(1)$$

Postać Newtona rozwiązania interpolacji Hermite'a

Jeżeli znane są wartości  $f^{(i)}(x_j), j = 0, 1, ..., n, i = 0, 1, ..., m_j - 1$ , gdzie  $x_0 < 1$  $x_1 < x_2 < \ldots < x_n$ oraz $t_0 \leqslant t_1 \leqslant t_2 \leqslant \ldots \leqslant t_m,$ dane są przez  $t_0 = t_1 = \ldots = t_m$  $t_{m_0-1}=x_0,\,t_{m_0}=t_{m_0+1}=...=t_{m_0+m_1-1}=x_1,...,$  (czyli wśród punktów  $t_i$ węzeł  $x_i$  występuje  $m_i$  razy), to wielomian

$$P(x) = f[t_0] + f[t_0,t_1](x-t_0) + f[t_0,t_1,t_2](x-t_0)(x-t_1) + f[t_0,t_1,...,t_m](x-t_0)...(x-t_{m-1})$$
spełnia warunki

$$P^{(i)}(x_j) = f^{(i)}(x_j), j = 0, 1, ..., n, i = 0, 1, ..., m_j - 1$$

Współczynniki dla  $P(x) = b_0 + b_1 x + b_2 x^2 + b_3 x^2 (x-1)$  wyznaczymy za pomocą ilorazów różnicowych:

$$f[0] = -1$$

$$f[0,0]=1$$

$$f[0,0,1]=-1$$

$$f[0,0,1,1]=2$$

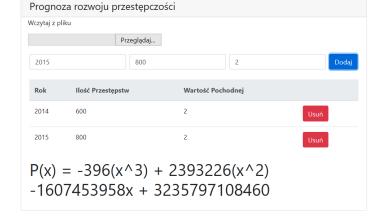
Otrzymujemy:  $b_0 = -1, b_1 = 1, b_2 = -1, b_3 = 2$ , a szukanym wielomianem jest:  $P(x) = 2x^3 - 5x^2 + x - 1$ 

#### 1.1 Opis implementacji algorytmu realizującego metodę

Algorytm jest algorytmem rekurencyjnym, który dla różnych danych wejściowych rozgałęzia się na dwie ścieżki w zależności czy  $t_0 = t_m$ .

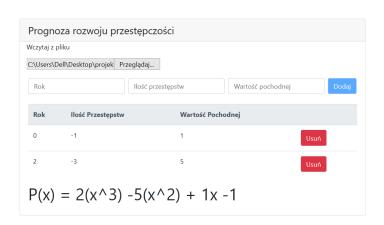
## 2 Działanie programu

# Interpolacja Wielomianowa - metoda Hermite'a



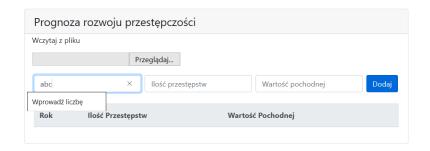
Rysunek 1: Dodawanie wartości

#### Interpolacja Wielomianowa - metoda Hermite'a



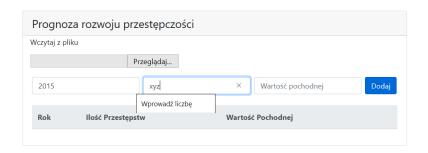
Rysunek 2: Dodawanie wartości z pliku

### Interpolacja Wielomianowa - metoda Hermite'a



Rysunek 3: Wprowadzenie niepoprawnego roku

### Interpolacja Wielomianowa - metoda Hermite'a



Rysunek 4: Wprowadzenie niepoprawnej ilości przestępstw

## 3 Opis programu

#### 3.1 Struktura programu

Program napisany jest w JavaScript, wykorzystuje HTML, CSS, Bootstrap. W celu realizowania zadania program wykorzystuje listy, funkcje rekurencyjne oraz funkcje silni.

#### 3.2 Wejście-wyjście

Wejście programu to dane w postaci  $x_i$ ,  $f(x_i)$ ,  $f'(x_i)$ , a po dodaniu dwóch par zmiennych w owej postaci program na wyjściu wyświetla wyliczony wielo-

mian. Dane mogą zostać wprowadzone przez formularz przeglądarki lub poprzez wczytanie pliku. Aby przeciwdziałać wprowadzeniu nieprawidłowych danych w formularzu ustawione są typy pól, które uniemożliwiają wpisywanie niepoprawnych danych. Dane dodawane są po wciśnięciu przycisku, a wyświetlane, kiedy po operacji dodawania liczba danych jest równa  $2 * (x_i, f(x_i), f'(x_i))$ .

#### 3.3 Kod programu

#### 3.3.1 JavaScript

```
let form = document.querySelector('.form-inline')
   let year = document.getElementById('rok')
   let crime_number = document.getElementById('iloscPrzestepstw')
   let derivative = document.getElementById('pochodna')
   let list = document.getElementById('tableBody')
   var data = []
   let polyniomaBox = document.getElementById('polyniomial')
   form.addEventListener('submit', createList);
q
   let input = document.querySelector('fileselector')
10
11
   function readSingleFile(e) {
       document.getElementById("add").disabled = true;
12
13
       var file = e.target.files[0];
       if (!file) {
14
15
         return:
16
       var reader = new FileReader();
17
       reader.onload = function(e){
18
19
            var content = e.target.result;
20
            displayContents(content);
21
22
23
       reader.readAsText(file);
24
25
26
27
     function displayContents(content){
28
       var lines = content.split(/\r\n|\n/);
29
30
       for (i=0; i<lines.length;i++){</pre>
            var number = lines[i].split(" ");
31
32
           var year = number[0];
33
           var crime_number = number[1];
34
           var deriv = number[2];
35
            createListFromFile(year, crime_number, deriv);
36
37
38
39
40
   document.getElementById('file-selector').addEventListener('change',
41
        readSingleFile, false);
42
43
   function deleteRow(o){
44
```

```
45
        var p = o.parentNode.parentNode;
46
        var year = p.childNodes[0].firstChild.nodeValue
47
        var crime_number = p.childNodes[1].firstChild.nodeValue
48
        var derivative = p.childNodes[2].firstChild.nodeValue
49
        for(i=0; i<data.length; i++){</pre>
50
            if(data[i][0]==year && data[i][1]==crime_number && data[i
                ][2] == derivative) {
51
                data.splice(i,1)
52
                console.log("Deleted values: ["+year+", "+crime_number+
                    derivative+"]")
53
           }
       }
54
55
56
        p.parentNode.removeChild(p);
57
        console.log("Array after deletion: "+data)
58
        if (data.length == 0){
59
            document.getElementById("add").disabled = false;
60
61
62
63
64
   function createListFromFile(year, crime_number, derivative){
65
66
         if(year != '' && crime_number !=''){
67
             let list_row = document.createElement('tr')
             \label{list_row.innerHTML} \verb| list_row.innerHTML = ""+year + ""+""+crime_number" | |
68
                 +""+""+""+"<button
                 type=\"button\" class=\"btn btn-danger\" onclick=\"
                 deleteRow(this)\">Usuń</button>"+""
69
             list.appendChild(list_row)
70
             var temp_table =[]
             if (derivative != ""){
71
72
                temp_table=derivative.split(" ")
73
             }
74
             var temp = []
75
             if (temp_table.length != 0){
76
             for(var element of temp_table){
77
             temp.push(parseInt(element))
78
             }
79
             data.push([parseInt(year), parseInt(crime_number), temp])
80
             }
81
             else{
                 data.push([parseInt(year), parseInt(crime_number)])
82
83
84
             console.log("Added values: ["+year+", "+crime_number+", "+
85
                 derivative+"]")
86
             console.log("Array after addition: "+data);
87
88
             if (data.length == 2)
89
                 interpolation(data)
90
         }
91
92
93
94
95
```

```
96
97
98
    function createList(e){
99
    e.preventDefault();
100
101
       // console.log(typeof year.value)
if(year != '' && crime_number !=''){
102
103
             let list_row = document.createElement('tr')
             list_row.innerHTML=""+year.value+""+""+
104
                 crime_number.value+""+""+derivative.value+"//pre>
                 td>"+""+" <button type=\"button\" class=\"btn btn-
                 danger\" onclick=\"deleteRow(this)\">Usuń</button>"+"
105
             list.appendChild(list_row)
106
             var temp_table =[]
             if (derivative.value != ""){
107
                temp_table=derivative.value.split(" ")
108
109
110
             var temp = []
             if (temp_table.length != 0){
111
112
             for(var element of temp_table){
113
             temp.push(parseInt(element))
114
115
             data.push([parseInt(year.value), parseInt(crime_number.
                 value), temp])
             }
116
117
             else{
118
                 data.push([parseInt(year.value), parseInt(crime_number.
                     value)])
119
120
121
             console.log("Added values: ["+year.value+", "+crime_number.
                 value+", "+derivative.value+"]")
122
             console.log("Array after addition: "+data);
123
124
             if (data.length==2)
125
                 interpolation(data)
126
127
        }
128
129
130
131
132
133
    function interpolation(data){
134
         var data_cloning = []
135
         for (var ele of data){
136
             var j=0
137
             if (ele.length==3){
             j = ele.length-1+(ele[2].length-1)
138
139
140
             else{
             j = ele.length-1
141
142
             while(j>0){
143
144
                 data_cloning.push(ele)
145
```

```
146
147
            }
148
        }
149
        for (var ele of data_cloning){
150
             console.log(ele)
151
        }
        b=[]
152
153
        var i=data_cloning.length-1
154
        var len=0
        while(i >= len){
155
156
             b.push(iloraz_roznicowa(data_cloning.slice(0,data_cloning.
                 length-i)))
157
158
159
        display_status(b,data_cloning)
160
161
162
163
    function display_status(b,t){
        j=0
164
165
        for (var i of b){
166
             console.log("b["+ j+++"] = "+i)
167
        lista_znakow=[]
168
169
        if ((b[0]-b[1]*t[0][0]+b[2]*t[0][0]*t[1][0]+b[3]*t[0][0]*t
             [2][0]-
170
             b[3]*t[0][0]*t[1][0]*t[2][0]) >= 0){
             lista_znakow.push("+")
171
172
        }
173
        else{
174
             lista_znakow.push("")
175
176
        if (b[1]-b[1]*t[1][0]-b[2]*t[0][0]+b[3]*t[1][0]*t[2][0] >= 0){
177
             lista_znakow.push("+")
178
        }
179
        else{
180
            lista_znakow.push("")
181
182
        if (b[2]-t[2][0]*b[3]-b[3]*t[1][0]>=0){
183
             lista_znakow.push("+")
        }
184
185
        else{
186
             lista_znakow.push("")
187
188
        var polynomial = "P(x) = "+(b[3])+"(x^3) "+lista_znakow[2]+" "
189
             +(b[2]-t[2][0]*b[3]-b[3]*t[1][0]-t[0][0]*b[3])+"(x^2)"+
            lista_znakow[1]+
         " "+(b[1]-b[1]*t[1][0]-b[2]*t[0][0]+b[3]*t[1][0]*t[2][0])+"x "+
190
             lista_znakow[0]+" "+(b[0]-b[1]*t[0][0]+b[2]*t[0][0]*t
             [1][0]+b[3]*t[0][0]*t[2][0]-
191
        b[3]*t[0][0]*t[1][0]*t[2][0])
192
193
        printPolynomial(polynomial)
194
195
196
```

```
197
198
199
    function iloraz_roznicowa(data){
200
         if (data.length >=2 ){
201
             if (data[0][0] == data[data.length -1][0]){
202
                 return(data[0][2][data.length-2]/factorial(data.length
                     -1))
203
             }
204
             else{
205
                 return iloraz_roznicowy_rozny(data)
206
207
        }
208
        else{
209
             return data[0][1]
210
211
212
    function iloraz_roznicowy_rozny(data){
213
        return (iloraz_roznicowa(data.slice(1)) - iloraz_roznicowa(data
             .slice(0,data.length-1)))/(data[data.length-1][0]-data
             [0][0]
214
215
216
    function factorial(n){
217
        if(n==0){
218
             return 1
219
220
        else{
221
             return n * factorial(n-1)
222
223
224
225
    function printPolynomial(polyniomial){
226
        polyniomaBox.innerHTML="<h1>"+polyniomial+"</h1>"
227
```

#### 3.3.2 HTML

```
<!DOCTYPE html>
1
2
   <html lang="en">
3
   <head>
       <meta charset="UTF-8">
4
5
        <meta name="viewport" content="width=device-width, initial-</pre>
            scale=1.0">
6
       <link rel="stylesheet" href="css/bootstrap.css">
7
       <title>Interpolacja Wielomianowa</title>
8
   </head>
10
   <body>
11
       <nav class="navbar navbar-dark bg-dark">
12
            <span class="navbar-text text-white mx-auto h2">
               Interpolacja Wielomianowa - metoda Hermite'a
13
14
            </span>
15
       </nav>
16
17
        <div class="card mt-5 w-50 mx-auto">
            <div class="card-header h4">
18
```

```
19
              Prognoza rozwoju przestępczości
20
             </div>
21
            <div class="mx-3">
22
              Wczytaj z pliku
              <input type="file" id="file-selector" accept=".txt, .</pre>
23
                  json">
            </div>
24
25
           <div class="card-body">
26
              <form class="form-inline">
                  <label class="sr-only" for="rok">Rok</label>
27
                  <input type="number" class="form-control mb-2 mr-sm
-2" id="rok" placeholder="Rok">
28
29
30
                  <label class="sr-only" for="iloscPrzestepstw">Ilość
                       przestępstw</label>
31
                  <input type="number" class="form-control mb-2 mr-sm</pre>
                      -2" id="iloscPrzestepstw" placeholder="Ilość
                      przestępstw">
32
33
                  <label class="sr-only" for="iloscPrzestepstw">
                      Wartość pochodnej pierwszego stopnia </label>
                  <input type="text" class="form-control mb-2 mr-sm</pre>
34
                      -2" id="pochodna" placeholder="Wartość
                      pochodnej">
35
                  <button type="submit" id="add" class="btn btn-</pre>
36
                      primary mb-2">Dodaj </button>
37
              </form>
              38
39
                  <thead class="thead-light">
40
                      Rok
41
42
                      Ilość Przestępstw
                      Wartość Pochodnej
43
44
                      45
                    46
                  </thead>
47
                  48
49
                  50
                <div id="polyniomial"></div>
51
           </div>
52
53
         </div>
54
         <script src="script.js"></script>
   </body>
55
   </html>
```

#### 3.3.3 CSS - fragment

```
html {
  font-family: sans-serif;
  line-height: 1.15;
  -webkit-text-size-adjust: 100%;
  -webkit-tap-highlight-color: rgba(0, 0, 0, 0);
}
```

```
7
   article, aside, figcaption, figure, footer, header, hgroup, main,
      nav, section {
9
     display: block;
10 }
11
12
   body {
13
    margin: 0;
14
     font-family: -apple-system, BlinkMacSystemFont, "Segoe UI",
         Roboto, "Helvetica Neue", Arial, "Noto Sans", sans-serif,
         Apple Color Emoji", "Segoe UI Emoji", "Segoe UI Symbol", "
         Noto Color Emoji";
15
     font-size: 1rem;
16
     font-weight: 400;
     line-height: 1.5;
17
18
     color: #212529;
19
     text-align: left;
20
     background-color: #fff;
21 }
22
23
   [tabindex="-1"]:focus:not(:focus-visible) {
24
   outline: 0 !important;
25 }
26
27
   hr {
28
     box-sizing: content-box;
29
    height: 0;
30
    overflow: visible;
31 }
32
33 h1, h2, h3, h4, h5, h6 {
34
   margin-top: 0;
35
    margin-bottom: 0.5rem;
36 }
37
38
39
   margin-top: 0;
40
    margin-bottom: 1rem;
41
42
43 abbr[title],
44 abbr[data-original-title] {
45
   text-decoration: underline;
46
     -webkit-text-decoration: underline dotted;
47
     text-decoration: underline dotted;
     cursor: help;
48
49
     border-bottom: 0;
50
     -webkit-text-decoration-skip-ink: none;
51
     text-decoration-skip-ink: none;
52
53
54 address {
55
    margin-bottom: 1rem;
56
     font-style: normal;
     line-height: inherit;
57
58
59
```