**Príloha B - Kódy projektu**

V tejto prílohe sa nachádzajú úplne kódy programu MATLAB s výnimkou kódov aplikácii, kde budú uvedené len tie časti kódu, ktoré je možné upravovať (vid. kap. 1.5). Sem partia funkcie spätných volaní (z angl. Callbacks), pomocné funkcie (z angl. Functions) a vlastnosti aplikácie (z angl. Properties). Jazyk MATLAB, rovnako ako drvivá väčšina iných programovacích jazykov neumožňuje pomenovávať vystupujúce premenné v kóde s použitím znakov s diakritikou a tak isto aj niektorých špeciálnych znakov. Z tohto dôvodu sú všetky kódy písané v anglickom jazyku s komentármi v slovenskom jazyku. Komentáre sú v kóde označené zelenou farbou, resp. znakom percenta na začiatku riadkov.

**1. Kód funkcie *create\_values\_file***

1. function varargout = create\_values\_file(varargin)
2. % Vstupné argumenty funkcie:
3. % PATH = Cesta k súboru s príponou \*.CSV.
4. % STEP = Šírka krokov medzi jednotlivými meraniami v sekundách.
5. % UNITS = Jednotky, ktoré sa uložili spolu s číselnou hodnotou do jedného
6. % stĺpca a je potrebné ich odstrániť. Je potrebné ponechať len
7. % číselné údaje. Ak nemajú údaje žiadne jednotky, neuvádza sa
8. % nič.
9. PATH = varargin{1};
10. STEP = varargin{2};
11. UNITS = varargin{3};
12. % V tejto častí sa do príslušných premenných funkcie načíta \*.CSV súbor,
13. % počet jeho riadkov a numerický vyjadrené kľúčové časové dĺžky, ktoré sa
14. % využívajú v ďalších častiach tejto funkcie:
15. CLEAN = readtable(PATH);
16. ROWS\_ALL = size(CLEAN, 1);
17. HOUR = 60;
18. MINUTES = minutes(HOUR);
19. SECONDS = seconds(MINUTES);
20. STEP = seconds(STEP);
21. % Na tomto mieste sa overí, či je v súbore zaznamenaných aspoň toľko
22. % riadkov, koľko pri definovanej šírke kroku zodpovedá jednej hodine:
23. if ROWS\_ALL >= SECONDS
24. % Inicializačná časť ETL procesu. V tejto časti dochádza k úpravám
25. % súboru ako úprava názvu jednotlivých stĺpcov, zaokrúhľovanie časových
26. % údajov na celé sekundy, zlúčenie stĺpcov dátumu a času do jedného
27. % spoločného stĺpca, konverzia textového údajového typu numerických
28. % údajov na číselný údajový typ, odstránenie jednotiek veličiny,
29. % odstránenie duplicitných riadkov (ak také sú), odstránenie prázdnych
30. % riadkov a konverzia údajového typu table na timetable:
31. CLEAN.Properties.VariableNames = {'Date', 'Time', 'Value'};
32. CLEAN.Date = datetime(CLEAN.Date, 'InputFormat', "d.M.uuuu", 'Format', "d.M.uuuu");
33. CLEAN.Time = duration(CLEAN.Time, 'Format', "hh:mm:ss");
34. CLEAN.Datetime = CLEAN.Date + CLEAN.Time;
35. CLEAN = removevars(CLEAN, {'Date', 'Time'});
36. CLEAN.Datetime = datetime(CLEAN.Datetime, 'Format', "d. M. uuuu HH:mm:ss");
37. CLEAN.Datetime.Second = floor(CLEAN.Datetime.Second);
38. CLEAN.Value = replace(CLEAN.Value, " " + UNITS, "");
39. CLEAN.Value = replace(CLEAN.Value, ",", ".");
40. CLEAN.Value = str2double(CLEAN.Value);
41. CLEAN = rmmissing(CLEAN);
42. CLEAN = timetable(CLEAN.Datetime, CLEAN.Value);
43. CLEAN.Properties.VariableNames = {'Value'};
44. CLEAN = sortrows(CLEAN, 'Time');
45. CLEAN = retime(CLEAN, unique(CLEAN.Time));
46. ROWS\_CLEAN = size(CLEAN, 1);
47. EXCLUDED = CLEAN;
48. % V tejto časti funkcia odstráni krátky úsek merania na začiatku aj na
49. % konci a to tak, aby meranie začínalo od nultej minúty nasledujúcej
50. % hodiny a končilo poslednou minutou predposlednej hodiny merania.:
51. begin\_of\_file = CLEAN.Time.Minute(1:SECONDS) == 0 & ...
52. CLEAN.Time.Second(1:SECONDS) == 0;
53. end\_of\_file = CLEAN.Time.Minute(end - SECONDS:end) == 0 & ...
54. CLEAN.Time.Second(end - SECONDS:end) == 0;
55. begin\_index = find(begin\_of\_file, 1);
56. end\_index = size(CLEAN, 1) - (SECONDS - find(end\_of\_file, 1) + 2);
57. CLEAN = CLEAN(begin\_index:end\_index, :);
58. % V tejto časti funkcie dochádza k odstráneniu neúplných 1-hodinových
59. % častí od merania, kde bolo meranie prerušené a zaradí tieto úseky do
60. % sekcie vypustených údajov.:
61. start = (CLEAN.Time(1):MINUTES:CLEAN.Time(end))';
62. stop = ((CLEAN.Time(1) + MINUTES - STEP):MINUTES:CLEAN.Time(end))';
63. [~, start] = ismember(start, CLEAN.Time);
64. [~, stop] = ismember(stop, CLEAN.Time);
65. sample = table(start, stop);
66. for i = 1:size(sample, 1)
67. if (sample.start(i) == 0) && (sample.stop(i) == 0)
68. sample.start(i) = NaN;
69. end
70. end
71. sample(isnan(sample.start), :) = [];
72. for i = 1:size(sample, 1)
73. if sample.start(i) == 0
74. sample.start(i) = sample.stop(i - 1) + 1;
75. end
76. if sample.stop(i) == 0
77. if sample.start(i + 1) == 0
78. sample.stop(i) = sample.stop(i + 1);
79. sample.start(i + 1) = NaN;
80. else
81. sample.stop(i) = sample.start(i + 1) - 1;
82. end
83. end
84. end
85. sample(isnan(sample.start), :) = [];
86. sample.observations = abs(sample.stop - sample.start) + 1;
87. for i = 1:size(sample, 1)
88. sample.c1(i) = ~isregular(CLEAN(sample.start(i):sample.stop(i), []));
89. sample.c2(i) = sample.observations(i) ~= SECONDS;
90. sample.c3(i) = range(CLEAN.Value(sample.start(i):sample.stop(i))) == 0;
91. sample.test(i) = sample.c1(i) + sample.c2(i) + sample.c3(i);
92. end
93. sample = sortrows(sample, 'test', 'descend');
94. for i = 1:nnz(sample.test)
95. CLEAN.Value(sample.start(i):sample.stop(i)) = NaN;
96. end
97. nans = isnan(CLEAN.Value);
98. CLEAN(nans, :) = [];
99. % Funkcia vytvorí časovú tabuľku vyradených hodnôt metódou porovnávania
100. % vstupnej tabuľky s výstupnou tabuľkou.:
101. EXCLUDED = EXCLUDED(~ismember(EXCLUDED.Time, CLEAN.Time), :);
102. % Funkcia vypíše výsledné štatistiky ETL procesu:
103. ROWS\_AFTER = size(CLEAN, 1);
104. fprintf("Vytváranie bolo dokončené.\n");
105. fprintf("Celkový počet údajov:\t\t%d\n", ROWS\_ALL);
106. fprintf("Počet vyčistených údajov:\t%d\n", ROWS\_AFTER);
107. fprintf("Počet vylúčených údajov:\t%d\n", ROWS\_CLEAN - ROWS\_AFTER);
108. fprintf("Počet poškodených údajov:\t%d\n\n", ROWS\_ALL - ROWS\_CLEAN);
109. % Funkcia vyradí tento \*.CSV súbor z ETL procesu, pretože nemá dostatok
110. % meraní. V takomto prípade funkcia vráti prázdne výstupy:
111. else
112. CLEAN = [];
113. EXCLUDED = [];
114. fprintf("Nedostatok (%d/%d) meraní. Súbor bol vylúčený.\n\n", ROWS\_ALL, SECONDS);
115. end
116. % Výstupné argumenty:
117. % CLEAN = Časová tabuľka čistých údajov po úprave ETL procesom.
118. % EXCLUDED = Časová tabuľka vypustených údajov po úprave ETL procesom.
119. varargout{1} = CLEAN;
120. varargout{2} = EXCLUDED;
121. end

**2. Kód funkcie *create\_values***

1. function varargout = create\_values(varargin)
2. % Vstupné argumenty:
3. % PATH = Cesta k súboru s príponou \*.CSV.
4. % STEP = Šírka krokov medzi jednotlivými meraniami v sekundách.
5. % UNITS = Jednotky, ktoré sa uložili spolu s číselnou hodnotou do
6. % jedného stĺpca a je potrebné ich odstrániť, pretože je
7. % potrebné ponechať len číselné údaje. Ak nemajú údaje žiadne
8. % jednotky, neuvádza sa nič.
9. % PROGRESS = Je potrebné uviesť objekt dialógového okna v aplikácii, pokiaľ
10. % bola táto funkcia zavolaná z aplikácie.
11. DATA\_PATH = varargin{1};
12. STEP = varargin{2};
13. UNITS = varargin{3};
14. if nargin == 4
15. PROGRESS = varargin{4};
16. end
17. % Inicializačná časť funkcie:
18. % V tejto časti sa spusti časovač a zaznamenávanie všetkého, čo sa
19. % prostredníctvom funkcie fprintf ocitne v okne príkazového riadku. Záznam
20. % sa následne uloží pod názvom log\_create\_values.txt do zložky Proces ETL.:
21. tic;
22. SCRIPT\_PATH = split(string(mfilename('fullpath')), "\");
23. SCRIPT\_PATH = join(SCRIPT\_PATH(1:end - 1), "\");
24. BASE\_PATH = split(SCRIPT\_PATH, "\");
25. BASE\_PATH = join(BASE\_PATH(1:end - 2), "\");
26. LOG\_PATH = BASE\_PATH + "\2. Pripravené údaje\Proces ETL\";
27. diary (LOG\_PATH + "\log\_create\_values.txt");
28. fprintf("Dátum:\t" + datestr(now, "dd.mm.yyyy") + "\n");
29. fprintf("Čas:\t" + datestr(now, "HH:MM") + "\n");
30. fprintf("-----VSTUPNÉ-ARGUMENTY-----\n");
31. fprintf("Cesta k priečinku:\t""%s""\n", DATA\_PATH);
32. fprintf("Krokovanie (s):\t\t%d\n", STEP);
33. fprintf("Jednotky:\t\t\t""%s""\n", UNITS);
34. fprintf("---------------------------\n");
35. % Funkcia vytvorí zoznam všetkých \*.CSV súborov v zadanej zložke a vypíše
36. % ich počet do príkazového riadku.:
37. FOLDER = datastore(DATA\_PATH + "\\*.csv");
38. FILE\_PATHS = FOLDER.Files;
39. FILE\_COUNT = size(FILE\_PATHS, 1);
40. fprintf("V priečinku sa našlo %d platných súborov.\n\n", FILE\_COUNT);
41. % Z textových reťazcov celých ciest k jednotlivým súborom, funkcia
42. % vyextrahuje len názvy súborov.:
43. FILE\_NAMES = split(FILE\_PATHS, "\");
44. FILE\_NAMES = string(FILE\_NAMES(:, end));
45. % Funkcia vyhradí priestor na RAM pamäti pre údaje, ktoré sa budú v
46. % nasledujúcom cykle načítavať.:
47. data = cell(4, 1);
48. % Hlavný cyklus for:
49. % Cyklus postupne zavolá pre každý \*.CSV súbor v priečinku funkciu
50. % create\_values\_file a zlúči jednotlivé súbory do jednej časovej tabuľky.
51. for i = 1:FILE\_COUNT
52. % Funkcia vypíše do príkazového riadku informáciu o názve súboru, ktorý
53. % vstupuje do ETL procesu.:
54. MSG = "Pripravuje sa " + i + ". súbor z celkového počtu " ...
55. + FILE\_COUNT + "...";
56. fprintf("(%s)\n%s\n", FILE\_NAMES(i), MSG);
57. if exist("PROGRESS", 'var')
58. PROGRESS.Value = i / (FILE\_COUNT + 11);
59. PROGRESS.Message = MSG;
60. end
61. % Funkcia postupne spája časové tabuľky, ktoré sú výstupmi funkcie
62. % create\_values\_file do jednej tabuľky.:
63. [data{3}, data{4}] = create\_values\_file(FILE\_PATHS{i}, STEP, UNITS);
64. data{1} = [data{1}; data{3}];
65. data{2} = [data{2}; data{4}];
66. data{3} = [];
67. data{4} = [];
68. end
69. % V tejto časti sa riadky výsledných časových tabuliek zoradia podľa času a
70. % zaokrúhľujú sa merané hodnoty na prvé desatinné miesto.:
71. if size(data{1}, 1) > 0
72. % Časová tabuľka čistých údajov:
73. fprintf("Vytvára sa časová tabuľka čistých údajov...\n");
74. CLEAN = sortrows(data{1}, 'Time');
75. CLEAN.Value = round(CLEAN.Value, 1);
76. CLEAN = retime(CLEAN, unique(CLEAN.Time));
77. if size(data{2}, 1) > 0
78. % Časová tabuľka vypustených údajov:
79. fprintf("Vytvára sa časová tabuľka vypustených údajov...\n");
80. EXCLUDED = sortrows(data{2}, "Time");
81. EXCLUDED.Value = round(EXCLUDED.Value, 1);
82. EXCLUDED = retime(EXCLUDED, unique(EXCLUDED.Time));
83. end
84. else
85. % Pokiaľ nebol nájdený žiadny súbor s viac ako 3600 meraniami, funkcia
86. % vráti prázdne časové tabuľky.:
87. fprintf("Nebol nájdený žiadny súbor s viac ako 3600 meraniami.\n");
88. CLEAN = [];
89. EXCLUDED = [];
90. end
91. % Ukončovacia časť funkcie:
92. % V tejto časti funkcia vráti všetky výstupné argumenty a vypíše dĺžku
93. % trvania výpočtov tejto funkcie. Zároveň sa ukončí a uloží záznam do
94. % textového súboru log\_create\_values.txt.:
95. fprintf("Uplynutý čas: %s.\n", datestr(seconds(toc), "HH:MM:SS"));
96. fprintf("Vytváranie tabuľky bolo úspešne dokončené.\n");
97. fprintf("===========================\n");
98. diary off;
99. % Výstupné argumenty funkcie:
100. % CLEAN = Časová tabuľka čistých údajov po úprave ETL procesom.
101. % EXCLUDED = Časová tabuľka vypustených údajov po úprave ETL procesom.
102. % FILE\_COUNT = Počet \*.CSV súborov v priečinku.
103. varargout{1} = CLEAN;
104. varargout{2} = EXCLUDED;
105. varargout{3} = FILE\_COUNT;
106. end

**3. Kód funkcie *create\_missing***

1. function varargout = create\_missing(varargin)
2. % Vstupné argumenty funkcie:
3. % CLEAN = Časová tabuľka čistých údajov po úprave ETL procesom.
4. % RECORD\_STEP = Interval krokovania meracieho prístroja v sekundách.
5. CLEAN = varargin{1};
6. RECORD\_STEP = varargin{2};
7. % Inicializačná časť funkcie:
8. % V tejto časti sa spusti časovač a zaznamenávanie všetkého, čo sa
9. % prostredníctvom funkcie fprintf ocitne v okne príkazového riadku. Záznam
10. % sa následne uloží pod názvom log\_create\_missing.txt do zložky Proces
11. % ETL.:
12. tic;
13. SCRIPT\_PATH = split(string(mfilename('fullpath')), "\");
14. SCRIPT\_PATH = join(SCRIPT\_PATH(1:end - 1), "\");
15. BASE\_PATH = split(SCRIPT\_PATH, "\");
16. BASE\_PATH = join(BASE\_PATH(1:end - 2), "\");
17. LOG\_PATH = BASE\_PATH + "\2. Pripravené údaje\Proces ETL\";
18. diary (LOG\_PATH + "log\_create\_missing.txt");
19. fprintf("Dátum:\t" + datestr(now, "dd.mm.yyyy") + "\n");
20. fprintf("Čas:\t" + datestr(now, "HH:MM") + "\n");
21. fprintf("-----VSTUPNÉ-ARGUMENTY-----\n");
22. fprintf("Tabuľka values\_clean:\t%s\n", inputname(1));
23. fprintf("Krokovanie (s):\t\t\t%d\n", RECORD\_STEP);
24. fprintf("---------------------------\n");
25. fprintf("Vytvára sa tabuľka...\n");
26. % Vytvorí prázdne tabuľky s názvami MISSING (výstup tejto funkcie) a temp
27. % (pomocná tabuľka pre výpočty).:
28. STEP = seconds(RECORD\_STEP);
29. MISSING = table;
30. temp = table;
31. % Pomocná tabuľka pre tvorbu hlavnej výstupnej tabuľky MISSING:
32. % Pomocou metódy porovnávania rozdielov v čase medzi riadkami, v ktorých sú
33. % uložené údaje o dátume a čase, funkcia zistí interval chýbajúcich riadkov
34. % z hľadiska času.:
35. temp.time = datetime(CLEAN.Time(1):STEP:CLEAN.Time(end))';
36. temp.membership = ismember(temp.time, CLEAN.Time);
37. temp.diff = [0; diff(temp.membership)];
38. temp(temp.diff == 0, :) = [];
39. temp = sortrows(temp, "diff", "ascend");
40. % Hlavná výstupná tabuľka MISSING:
41. % V prvom stĺpci sa budú nachádzať dátumy a časy, kedy meranie bolo z
42. % neznámeho dôvodu pozastavené. V druhom stĺpci sa budú nachádzať dátumy a
43. % časy, kedy meranie pokračuje. V treťom stĺpci sa budú nachádzať numerické
44. % hodnoty predstavujúce počet chýbajúcich hodín, pretože chýbať môžu len
45. % presné násobky hodín po ETL procese spracovania zozbieraných údajov.:
46. MISSING.Begin = temp.time(1:(size(temp, 1) / 2));
47. MISSING.End = temp.time(((size(temp, 1) / 2) + 1):end);
48. MISSING.Hours = hours(MISSING.End - MISSING.Begin);
49. MISSING = sortrows(MISSING, "Hours", "ascend");
50. % Ukončovacia časť funkcie:
51. % V tejto časti funkcia vráti všetky výstupné argumenty a vypíše dĺžku
52. % trvania výpočtov tejto funkcie. Zároveň sa ukončí a uloží záznam do
53. % textového súboru log\_create\_missing.txt.:
54. fprintf("Uplynutý čas: %s.\n", datestr(seconds(toc), 'HH:MM:SS'));
55. fprintf("Vytváranie tabuľky bolo úspešne dokončené.\n");
56. fprintf("===========================\n");
57. diary off;
58. beep;
59. % Výstupné argumenty funkcie:
60. % MISSING = Tabuľka záznamov chýbajúcich hodnôt, ktoré boli vypustené
61. % ETL procesom z analýzy údajov.
62. varargout{1} = MISSING;
63. end

**4. Kód funkcie *create\_filled***

1. function varargout = create\_filled(varargin)
2. % Vstupné argumenty funkcie:
3. % MISSING = Tabuľka záznamov chýbajúcich hodnôt, ktoré boli vypustené
4. % ETL procesom z analýzy údajov.
5. % EXCLUDED = Časová tabuľka vypustených údajov po úprave ETL procesom.
6. % RECORD\_STEP = Interval krokovania meracieho prístroja v sekundách.
7. % TOLERANCE = Numerická hodnota, ktorá predstavuje hornú hranicu počtu
8. % hodín vrátane, kedy ešte algoritmus tejto funkcie vyplní
9. % chýbajúce riadky údajov.
10. MISSING = varargin{1};
11. EXCLUDED = varargin{2};
12. RECORD\_STEP = varargin{3};
13. TOLERANCE = varargin{4};
14. % Inicializačná časť funkcie:
15. % V tejto časti sa spusti časovač a zaznamenávanie všetkého, čo sa
16. % prostredníctvom funkcie fprintf ocitne v okne príkazového riadku. Záznam
17. % sa následne uloží pod názvom log\_create\_filled.txt do zložky Proces ETL.:
18. tic;
19. SCRIPT\_PATH = split(string(mfilename('fullpath')), "\");
20. SCRIPT\_PATH = join(SCRIPT\_PATH(1:end - 1), "\");
21. BASE\_PATH = split(SCRIPT\_PATH, "\");
22. BASE\_PATH = join(BASE\_PATH(1:end - 2), "\");
23. LOG\_PATH = BASE\_PATH + "\2. Pripravené údaje\Proces ETL\";
24. diary (LOG\_PATH + "log\_create\_filled.txt");
25. fprintf("Dátum:\t" + datestr(now, "dd.mm.yyyy") + "\n");
26. fprintf("Čas:\t" + datestr(now, "HH:MM") + "\n");
27. fprintf("-----VSTUPNÉ-ARGUMENTY-----\n");
28. fprintf("Tabuľka missing:\t\t\t%s\n", inputname(1));
29. fprintf("Tabuľka values\_excluded:\t%s\n", inputname(2));
30. fprintf("Tolerancia (h):\t\t\t\t%d\n", TOLERANCE);
31. fprintf("---------------------------\n");
32. fprintf("Vytvára sa tabuľka...\n");
33. % Funkcia načíta tabuľku MISSING, ktorá je výstupom funkcie create\_missing.
34. % Funkcia zoradí tabuľku podľa počtu chýbajúcich hodín a na základe počtu
35. % riadkov tabuľky MISSING funkcia urči počet cyklov nasledujúcej časti
36. % kódu.:
37. MISSING = sortrows(MISSING, "Hours", 'ascend');
38. LENGHT = size(MISSING, 1);
39. STEP = seconds(RECORD\_STEP);
40. FILLED = [];
41. % V tejto časti kódu, postupne od prvého riadku tabuľky MISSING zbehne
42. % hlavný cyklus for, ktorý pre každý neúplný N-hodinový interval
43. % (pričom N = TOLERANCE), dopočíta pomocou metódy automatického dopĺňania
44. % chýbajúcich údajov a pomocou pohyblivého priemeru vypočítaného z
45. % okolitých údajov doplní tieto medzery v riadkoch.:
46. for i = 1:LENGHT
47. % Funkcia odčíta hranice intervalu chýbajúcich hodnôt v danom riadku
48. % tabuľky MISSING. Na základe týchto údajov, funkcia vyextrahuje z
49. % časovej tabuľky EXCLUDED všetky namerané údaje v tomto intervale,
50. % ktoré sú pochopiteľne neúplné. V tejto časti sa stanoví počet
51. % chýbajúcich riadkov, počet predpokladaných riadkov a počet aktuálnych
52. % riadkov.:
53. BEGIN = MISSING.Begin(i);
54. END = MISSING.End(i);
55. HOURS = MISSING.Hours(i);
56. old\_data = EXCLUDED(timerange(BEGIN, END), :);
57. REGULAR\_TIMES = datetime(BEGIN:STEP:END - seconds(1))';
58. old\_data = retime(old\_data, REGULAR\_TIMES);
59. MISSING = sum(isnan(old\_data.Value));
60. EXPECTED = seconds(hours(HOURS)) / RECORD\_STEP;
61. ACTUAL = EXPECTED - MISSING;
62. % Úlohou tejto časti kódu je doplniť neúplné časové tabuľky pomocou
63. % metódy pohyblivého priemeru a to v závislosti od predpokladanej
64. % príčiny absencie týchto údajov. Keďže nevieme s istotou určiť
65. % príčinu, môžeme sa len domnievať na základe počtu chýbajúcich
66. % hodnôt aj to len v nieko¾kých konkrétnych prípadoch.:
67. MISSING.OLD\_Data{i} = old\_data;
68. MISSING.OLD\_Missing(i) = MISSING;
69. MISSING.OLD\_Missing\_Percents(i) = round(MISSING / EXPECTED, 4) \* 100;
70. if HOURS <= TOLERANCE
71. % Prípad č.1:
72. % Prípad, kedy nebola v danom časovom intervale nameraná ani jedná
73. % hodnota. V takomto prípade nie je čo dopĺňať a úsek ostáva
74. % prázdny.:
75. if ACTUAL == 0
76. MISSING.NEW\_Data{i} = [];
77. MISSING.NEW\_Missing(i) = EXPECTED;
78. % Prípad č.2:
79. % Prípad, kedy sa všetky namerané hodnoty rovnajú jednému a tomu
80. % istému číslu a zároveň nebolo meranie počas tohto časového úseku
81. % vynechané. V takomto prípade sa do tabuľky MISSING zapíše k
82. % tomuto záznamu nereálna hodnota -1 do stĺpca s počtom vynechaných
83. % hodnôt. Slúži len na odlíšenie pre používateľa, že sa jedná o
84. % tento špecifický prípad.:
85. elseif ACTUAL == EXPECTED && range(old\_data.Value) == 0
86. MISSING.NEW\_Data{i} = [];
87. MISSING.NEW\_Missing(i) = -1;
88. MISSING.NEW\_Missing(i) = EXPECTED;
89. % Prípad č.3:
90. % Prípad podobný prípadu č.2 avšak s tým rozdielom, že meraný
91. % signál nie je konštantný. Ide o špecifický prípad, kedy bol
92. % (pravdepodobne) bezchybný časový úsek vyhodnotený ako chybný.
93. % Keďže takýto časový úsek nemá žiadne chýbajúce riadky, funkcia
94. % nevykoná žiadne zmeny a posunie tento úsek do tabuľky FILLED bez
95. % úprav.:
96. elseif ACTUAL == EXPECTED && range(old\_data.Value) > 0
97. MISSING.NEW\_Data{i} = old\_data;
98. MISSING.NEW\_Missing(i) = sum(isnan(MISSING.NEW\_Data{i}.Value));
99. % Prípad č.4:
100. % Špecifický prípad, kedy nastáva zmena zimného času na letný čas.
101. % Opačný prípad kedy nastáva zmena letného času na zimný čas sa
102. % nemôže dostať do tabuľky neúplných časových úsekov, pretože do
103. % tejto tabuľky sa cez ETL proces dostanú len neúplné časové úseky.
104. % To znamená, že by tam musela byť zaznamenaná aspoň jedná hodnota
105. % v tom čase. Naopak v tomto prípade je vyhodnotený vždy 2-hodinový
106. % časový úsek, pretože posledná hodnota pred zmenou času sa
107. % zaznamená vždy o 02:00:01 a končí o 03:00:00. Vďaka posunu o
108. % jednu sekundu dozadu sa vyhodnotil tento 1-hodinový časový úsek
109. % ako 2-hodinový časový úsek, čo napomáha funkcii identifikovať
110. % tento špecificky prípad. Zároveň to znamená, že počet chýbajúcich
111. % hodnôt (3600 riadkov) je rovný počtu aktuálnych meraní (1. riadok
112. % + 3599 riadkov, ktoré nasledujú po hodinovej medzere). Podobne
113. % ako v druhom prípade je tento špecificky prípad označený
114. % nereálnou hodnotou počtu chýbajúcich riadkov a to -2.:
115. elseif HOURS == 2 && MISSING == ACTUAL && old\_data.Time.Month(1) == 3
116. MISSING.Hours(i) = 1;
117. old\_data(isnan(old\_data.Value), :) = [];
118. old\_data.Time(1) = old\_data.Time(2) - seconds(1);
119. MISSING.NEW\_Data{i} = old\_data;
120. MISSING.NEW\_Missing(i) = -2;
121. % Prípad č.5:
122. % Sem patria všetky ostatné prípady, ktoré je potrebné doplniť
123. % metódou pohyblivého priebehu. V tomto prípade sa kontroluje
124. % len podmienka, či počet riadkov je nenulový. Do výslednej časovej
125. % tabuľky sa posúva doplnený časový úsek.:
126. elseif ACTUAL > 0
127. RANGE = seconds(MISSING + 1);
128. MISSING.NEW\_Data{i} = fillmissing(old\_data, 'movmean', RANGE);
129. MISSING.NEW\_Data{i}.Value = round(MISSING.NEW\_Data{i}.Value, 1);
130. MISSING.NEW\_Missing(i) = sum(isnan(MISSING.NEW\_Data{i}.Value));
131. end
132. % Do výstupnej časovej tabuľky FILLED, pridá funkcia doplnený
133. % časový úsek.:
134. if i == 1
135. FILLED = MISSING.NEW\_Data{1};
136. else
137. FILLED = [FILLED; MISSING.NEW\_Data{i}]; %#ok<AGROW>
138. end
139. end
140. end
141. % Táto časť kódu uloží všetky grafické priebehy signálov, ktoré boli
142. % doplnené v závislosti od stanovenej tolerancie dopĺňania údajov.:
143. FIGURES\_PATH = BASE\_PATH + "\3. Analýza údajov\Chyby údajov\" ...
144. + "Automaticky upravené riadky\Obrázky\";
145. MISSING = sortrows(MISSING, {'Hours', 'Begin'}, 'ascend');
146. for i = 1:LENGHT
147. if MISSING.Hours(i) <= TOLERANCE
148. try
149. h = figure;
150. plot(h, MISSING.NEW\_Data{i}.Time, MISSING.NEW\_Data{i}.Value);
151. savefig(h, FIGURES\_PATH + i + ".fig");
152. close(h);
153. catch
154. close(h);
155. end
156. end
157. end
158. % Ukončovacia časť funkcie:
159. % V tejto časti funkcia vráti všetky výstupné argumenty a vypíše dĺžku
160. % trvania výpočtov tejto funkcie. Zároveň sa ukončí a uloží záznam do
161. % textového súboru log\_create\_filled.txt.:
162. fprintf("Uplynutý čas: %s.\n", datestr(seconds(toc), "HH:MM:SS"));
163. fprintf("Vytváranie tabuľky bolo úspešne dokončené.\n");
164. fprintf("===========================\n");
165. diary off;
166. beep;
167. % Výstupné argumenty funkcie:
168. % MISSING = Tabuľka záznamov chýbajúcich hodnôt, ktoré boli vypustené
169. % ETL procesom z analýzy údajov. Táto tabuľka je rozšírená o
170. % stĺpce vyradených hodnôt a stĺpce doplnených hodnôt.
171. % FILLED = Časová tabuľka doplnených údajov po úprave ETL procesom.
172. varargout{1} = MISSING;
173. varargout{2} = FILLED;
174. end

**5. Kód funkcie *create\_statistics***

1. function varargout = create\_statistics(varargin)
2. % Vstupné argumenty funkcie:
3. % CLEAN = Časová tabuľka čistých údajov.
4. % SAMPLE\_SIZE = Šírka časového intervalu, podľa ktorého sa budú vypočítavať
5. % štatistické parametre. Dĺžka je uvedená v minútach.
6. CLEAN = varargin{1};
7. SAMPLE\_SIZE = varargin{2};
8. % Inicializačná časť funkcie:
9. % V tejto časti sa spusti časovač a zaznamenávanie všetkého, čo sa
10. % prostredníctvom funkcie fprintf ocitne v okne príkazového riadku. Záznam
11. % sa následne uloží pod názvom log\_create\_statistics.txt do zložky Proces
12. % ETL.:
13. tic;
14. SCRIPT\_PATH = split(string(mfilename('fullpath')), "\");
15. SCRIPT\_PATH = join(SCRIPT\_PATH(1:end - 1), "\");
16. BASE\_PATH = split(SCRIPT\_PATH, "\");
17. BASE\_PATH = join(BASE\_PATH(1:end - 2), "\");
18. LOG\_PATH = BASE\_PATH + "\2. Pripravené údaje\Proces ETL\";
19. diary (LOG\_PATH + "log\_create\_statistics.txt");
20. fprintf("Dátum:\t" + datestr(now, "dd.mm.yyyy") + "\n");
21. fprintf("Čas:\t" + datestr(now, "HH:MM") + "\n");
22. fprintf("-----VSTUPNÉ-ARGUMENTY-----\n");
23. fprintf("Tabuľka values\_clean:\t%s\n", inputname(1));
24. fprintf("Šírka intervalu:\t\t%d\n", SAMPLE\_SIZE);
25. fprintf("---------------------------\n");
26. fprintf("Vytvára sa tabuľka...\n");
27. % Zistí počet riadkov časovej tabuľky čistých údajov a na základe toho
28. % vypočíta počet riadkov, ktorý je nevyhnutný pre tvorbu tabuľky
29. % štatistických parametrov:
30. SIZE\_CLEAN = size(CLEAN, 1);
31. SAMPLE\_SECONDS = seconds(minutes(SAMPLE\_SIZE));
32. LENGHT = floor(SIZE\_CLEAN / SAMPLE\_SECONDS);
33. % Matica údajových typov a názvov stĺpcov tabuľky štatistických parametrov.
34. % Nevyhnutná pri tvorbe, resp. definovaní akejkoľvek prázdnej tabuľky.:
35. SETUP = [
36. "datetime", "DateStart";...
37. "datetime", "DateStop";...
38. "double", "Mean";...
39. "double", "Mode";...
40. "double", "Median";...
41. "double", "Minimum";...
42. "double", "Maximum";...
43. "double", "Range";...
44. "double", "IQR";...
45. "double", "Variance";...
46. "double", "SD";...
47. "double", "RSD";...
48. "double", "Skewness";...
49. "double", "Kurtosis";...
50. "double", "Equality";...
51. "double", "SEM";...
52. "double", "ProbableError";...
53. "double", "AccuracyRate";...
54. ];
55. % Vytvorí sa prázdna tabuľka, do ktorej sa v ďalšej časti funkcie uložia
56. % štatistické parametre.:
57. STATS = table(...
58. 'Size', [LENGHT, size(SETUP, 1)],...
59. 'VariableTypes', SETUP(:, 1),...
60. 'VariableNames', SETUP(:, 2));
61. % Textový reťazec pre cyklus for, ktorý definuje hranice intervalu v
62. % tabuľke čistých údajov.:
63. LOCATION = "clean.Value(((i - 1) \* SAMPLE\_SECONDS) + 1:";
64. LOCATION = LOCATION + "((i - 1) \* SAMPLE\_SECONDS) + SAMPLE\_SECONDS)";
65. % Vypočíta všetky štatistické parametre pre každý interval tabuľky čistých
66. % údajov. Zároveň tieto vypočítané hodnoty ukladá do tabuľky štatistických
67. % parametrov.:
68. for i = 1:LENGHT
69. STATS.DateStart(i) = CLEAN.Time(((i - 1) \* SAMPLE\_SECONDS) + 1);
70. STATS.DateStop(i) = CLEAN.Time(((i - 1) \* SAMPLE\_SECONDS) + SAMPLE\_SECONDS);
71. STATS.Mean(i) = eval(sprintf("mean(%s)", LOCATION));
72. STATS.Mode(i) = eval(sprintf("mode(%s)", LOCATION));
73. STATS.Median(i) = eval(sprintf("median(%s)", LOCATION));
74. STATS.Minimum(i) = eval(sprintf("min(%s)", LOCATION));
75. STATS.Maximum(i) = eval(sprintf("max(%s)", LOCATION));
76. STATS.Range(i) = eval(sprintf("range(%s)", LOCATION));
77. STATS.IQR(i) = eval(sprintf("iqr(%s)", LOCATION));
78. STATS.Variance(i) = eval(sprintf("var(%s)", LOCATION));
79. STATS.SD(i) = eval(sprintf("std(%s)", LOCATION));
80. STATS.Skewness(i) = eval(sprintf("skewness(%s)", LOCATION));
81. STATS.Kurtosis(i) = eval(sprintf("kurtosis(%s)", LOCATION));
82. end
83. % Vypočíta a doplní tabuľku štatistických parametrov o ďalšie štatistické
84. % parametre, ktoré sú odvodené z vyššie vypočítaných parametrov.:
85. STATS.RSD = (STATS.SD ./ STATS.Mean) \* 100;
86. STATS.Equality = std([STATS.Mean, STATS.Mode, STATS.Median], [], 2);
87. STATS.SEM = STATS.SD / sqrt(SAMPLE\_SECONDS);
88. STATS.ProbableError = STATS.SD \* 0.6745;
89. STATS.AccuracyRate = 1 ./ (STATS.SD \* sqrt(2));
90. % Ukončovacia časť funkcie:
91. % V tejto časti funkcia vráti všetky výstupné argumenty a vypíše dĺžku
92. % trvania výpočtov tejto funkcie. Zároveň sa ukončí a uloží záznam do
93. % textového súboru log\_create\_statistics.txt.:
94. fprintf("Uplynutý čas: %s.\n", datestr(seconds(toc), "HH:MM:SS"));
95. fprintf("Vytváranie tabuľky bolo úspešne dokončené.\n");
96. fprintf("===========================\n");
97. diary off;
98. beep;
99. % Výstupné argumenty funkcie:
100. % STATS = Tabuľka štatistických parametrov.
101. varargout{1} = STATS;
102. end

**6. Kód funkcie *etl\_update***

1. function etl\_update(varargin)
2. % Vstupné argumenty funkcie:
3. % OLD\_DATA\_PATH = cesta k priečinku, v ktorom sa nachádzajú \*.csv súbory.
4. % STEP = interval krokovania meracieho prístroja v sekundách.
5. % UNITS = jednotky meraných veličín.
6. % TOLERANCE = počet maximálne tolerovaných hodín, ktoré budú
7. % automatický doplnené.
8. % PROGRESS = ak je táto funkcia spustená cez aplikáciu, tak vďaka
9. % tomuto argumentu sa bude zobrazovať v dialógovom okne
10. % progres výpočtov.
11. OLD\_DATA\_PATH = varargin{1};
12. STEP = varargin{2};
13. UNITS = varargin{3};
14. TOLERANCE = varargin{4};
15. if nargin == 5
16. PROGRESS = varargin{5};
17. end
18. % Táto časť kódu uloží do premennej s názvom BASE\_PATH cestu v ktorej je
19. % uložená táto funkcia.:
20. SCRIPT\_PATH = split(string(mfilename('fullpath')), "\");
21. SCRIPT\_PATH = join(SCRIPT\_PATH(1:end - 1), "\");
22. BASE\_PATH = split(SCRIPT\_PATH, "\");
23. BASE\_PATH = string(join(BASE\_PATH(1:end - 2), "\"));
24. % Táto časť kódu zavolá funkcie ETL procesu v príslušnom poradí:
25. % Najprv vytvorí časovú tabuľku s názvom values, kde budú uložené všetky
26. % údaje načítané z \*.CSV súborov po úprave ETL procesom. Následne tá istá
27. % funkcia s názvom create\_values uloží do premennej values\_excluded všetky
28. % údaje vypustené ETL procesom.:
29. [values\_clean, values\_excluded, file\_count] = ...
30. create\_values(OLD\_DATA\_PATH, STEP, UNITS, PROGRESS);
31. % Následne funkcia s názvom create\_missing vytvorí tabuľku s názvom
32. % missing, v ktorej budú uložené všetky údaje o chýbajúcich, resp.
33. % vypustených údajoch.:
34. if exist("PROGRESS", 'var')
35. pd\_state = file\_count + 1;
36. PROGRESS.Value = pd\_state / (file\_count + 11);
37. PROGRESS.Message = "Vytvára sa tabuľka s názvom missing...";
38. end
39. missing = create\_missing(values\_clean, STEP);
40. % Táto tabuľka bude následne rozšírená o niekoľko stĺpcov, v ktorých budú
41. % uložené vypustené údaje a tie iste údaje doplnené o chýbajúce hodnoty.
42. % Doplnené údaje budú uložené v časovej tabuľke s názvom values\_filled.:
43. if exist("PROGRESS", 'var')
44. pd\_state = pd\_state + 1;
45. PROGRESS.Value = pd\_state / (file\_count + 11);
46. PROGRESS.Message = "Dopĺňajú sa chýbajúce hodnoty...";
47. end
48. [values\_filled, missing] = ...
49. create\_filled(missing, values\_excluded, STEP, TOLERANCE);
50. % Táto časť kódu spája čisté a doplnené hodnoty a následne ich zotriedi
51. % podľa času.:
52. if exist("PROGRESS", 'var')
53. pd\_state = pd\_state + 1;
54. PROGRESS.Value = pd\_state / (file\_count + 11);
55. PROGRESS.Message = "Zlučujú sa čisté a doplnené hodnoty...";
56. end
57. values\_clean = [values\_clean; values\_filled];
58. values\_clean = sortrows(values\_clean, "Time");
59. % Táto časť kódu vytvorí tabuľky s názvom stats pre všetky definované
60. % časové intervaly uložené v premennej SIGNAL\_LENGTHS.:
61. SIGNAL\_LENGTHS = [10, 12, 15, 20, 30, 60];
62. SIGNAL\_LENGTHS\_STRINGS = strings(length(SIGNAL\_LENGTHS), 1);
63. idx = 0;
64. for i = SIGNAL\_LENGTHS
65. if exist("PROGRESS", 'var')
66. pd\_state = pd\_state + 1;
67. PROGRESS.Value = pd\_state / (file\_count + 11);
68. PROGRESS.Message = "Vytvára sa tabuľka štatistických " + ...
69. "parametrov pre " + i + "-minútové intervaly...";
70. end
71. idx = idx + 1;
72. SIGNAL\_LENGTHS\_STRINGS(idx) = i + " min";
73. eval("stats\_" + i + " = create\_statistics(values\_clean, i);");
74. end
75. % Táto časť kódu rozdelí údaje podľa rokov.:
76. if exist("PROGRESS", 'var')
77. pd\_state = pd\_state + 1;
78. PROGRESS.Value = pd\_state / (file\_count + 11);
79. PROGRESS.Message = "Rozdeľujú sa údaje podľa rokov...";
80. end
81. YEARS = unique(values\_clean.Time.Year);
82. YEARS\_STRINGS = string(YEARS);
83. for i = 1:size(YEARS, 1)
84. eval("values\_clean\_" + YEARS(i) + ...
85. " = values\_clean(values\_clean.Time.Year == YEARS(i), :);");
86. eval("values\_excluded\_" + YEARS(i) + ...
87. " = values\_excluded(values\_excluded.Time.Year == YEARS(i), :);");
88. if ~isempty(values\_filled)
89. eval("values\_filled\_" + YEARS(i) + ...
90. " = values\_filled(values\_filled.Time.Year == YEARS(i), :);");
91. end
92. for j = SIGNAL\_LENGTHS
93. eval("stats\_" + j + "\_" + YEARS(i) + ...
94. " = stats\_" + j + "(stats\_" + j + ...
95. ".DateStart.Year == YEARS(i), :);");
96. end
97. end
98. % Táto časť kódu uloží všetky údaje do príslušných priečinkov.:
99. if exist("PROGRESS", 'var')
100. pd\_state = pd\_state + 1;
101. PROGRESS.Value = pd\_state / (file\_count + 11);
102. PROGRESS.Message = "Ukladajú sa údaje...";
103. end
104. NEW\_DATA\_PATH = BASE\_PATH + "\2. Pripravené údaje\";
105. ANALYSIS\_PATH = BASE\_PATH + "\3. Analýza údajov\";
106. MISSING\_PATH = BASE\_PATH + "\3. Analýza údajov\Chyby údajov\" + ...
107. "Automaticky upravené riadky\";
108. for i = 1:size(YEARS, 1)
109. save(NEW\_DATA\_PATH + "stats\_" + YEARS(i) + ".mat", "stats\_" + ...
110. SIGNAL\_LENGTHS(1) + "\_" + YEARS(i), "-v7.3");
111. for j = 2:length(SIGNAL\_LENGTHS)
112. save(NEW\_DATA\_PATH + "stats\_" + YEARS(i) + ".mat", "stats\_" + ...
113. SIGNAL\_LENGTHS(j) + "\_" + YEARS(i), "-append");
114. end
115. save(NEW\_DATA\_PATH + "values\_clean\_" + YEARS(i) + ...
116. ".mat", "values\_clean\_" + YEARS(i), "-v7.3");
117. save(NEW\_DATA\_PATH + "values\_excluded\_" + YEARS(i) + ...
118. ".mat", "values\_excluded\_" + YEARS(i), "-v7.3");
119. if ~isempty(values\_filled)
120. save(NEW\_DATA\_PATH + "values\_filled\_" + YEARS(i) + ...
121. ".mat", "values\_filled\_" + YEARS(i), "-v7.3");
122. end
123. end
124. save(ANALYSIS\_PATH + "years.mat", "YEARS\_STRINGS", "-v7.3");
125. save(ANALYSIS\_PATH + "signal\_lengths.mat", "SIGNAL\_LENGTHS\_STRINGS", "-v7.3");
126. save(MISSING\_PATH + "missing.mat", "missing", "-v7.3");
127. end

**7. Kód funkcie *etl\_append***

1. function etl\_append(varargin)
2. % Vstupné argumenty funkcie:
3. % OLD\_DATA\_PATH = cesta k priečinku, v ktorom sa nachádzajú \*.csv súbory
4. % alebo cesta ku konkrétnemu \*.csv súboru.
5. % STEP = interval krokovania meracieho prístroja v sekundách.
6. % UNITS = jednotky meraných veličín.
7. % TOLERANCE = počet maximálne tolerovaných hodín, ktoré budú
8. % automatický doplnené.
9. % PROGRESS = ak je táto funkcia spustená cez aplikáciu, tak vďaka
10. % tomuto argumentu sa bude zobrazovať v dialógovom okne
11. % progres výpočtov.
12. OLD\_DATA\_PATH = varargin{1};
13. STEP = varargin{2};
14. UNITS = varargin{3};
15. TOLERANCE = varargin{4};
16. if nargin == 5
17. PROGRESS = varargin{5};
18. end
19. % Táto časť kódu uloží do premennej s názvom BASE\_PATH cestu v ktorej je
20. % uložená táto funkcia:
21. SCRIPT\_PATH = split(string(mfilename('fullpath')), "\");
22. SCRIPT\_PATH = join(SCRIPT\_PATH(1:end - 1), "\");
23. BASE\_PATH = split(SCRIPT\_PATH, "\");
24. BASE\_PATH = string(join(BASE\_PATH(1:end - 2), "\"));
25. % Táto časť kódu zavolá funkcie ETL procesu v príslušnom poradí:
26. % Najprv vytvorí časovú tabuľku s názvom values\_clean\_NEW, kde budú uložené
27. % všetky údaje načítané z \*.CSV súborov po úprave ETL procesom. Následne tá
28. % istá funkcia s názvom create\_values, resp. create\_values\_file uloží do
29. % premennej values\_excluded\_NEW všetky údaje vyradené ETL procesom.
30. % Podmienka: Ak je prvým vstupným argumentom cesta k \*.csv súboru, zavolá
31. % sa funkcia create\_values\_file.:
32. is\_file = contains(OLD\_DATA\_PATH, ".csv");
33. if is\_file
34. % Inicializačná časť funkcie create\_values\_file:
35. % V tejto časti sa spusti časovač a zaznamenávanie všetkého, čo sa
36. % prostredníctvom funkcie fprintf ocitne v okne príkazového riadku.
37. % Záznam sa následne uloží pod názvom log\_create\_values.txt do zložky
38. % Proces ETL.:
39. tic;
40. LOG\_PATH = BASE\_PATH + "\2. Pripravené údaje\Proces ETL\";
41. diary (LOG\_PATH + "\log\_create\_values.txt");
42. fprintf("Dátum:\t" + datestr(now, "dd.mm.yyyy") + "\n");
43. fprintf("Èas:\t" + datestr(now, "HH:MM") + "\n");
44. fprintf("-----VSTUPNÉ-ARGUMENTY-----\n");
45. fprintf("Cesta k súboru:\t\t""%s""\n", OLD\_DATA\_PATH);
46. fprintf("Krokovanie (s):\t\t%d\n", STEP);
47. fprintf("Jednotky:\t\t\t""%s""\n", UNITS);
48. fprintf("---------------------------\n");
49. % Volanie funkcie create\_values\_file.:
50. file\_count = 1;
51. [values\_clean\_NEW, values\_excluded\_NEW] = ...
52. create\_values\_file(OLD\_DATA\_PATH, STEP, UNITS);
53. % Ukončovacia časť funkcie create\_values\_file:
54. % V tejto časti funkcia vráti všetky výstupné argumenty a vypíše dĺžku
55. % trvania výpočtov tejto funkcie. Zároveň sa ukončí a uloží záznam do
56. % textového súboru log\_create\_values.txt.:
57. fprintf("Uplynutý čas: %s.\n", datestr(seconds(toc), "HH:MM:SS"));
58. fprintf("Vytváranie tabuľky bolo úspešne dokončené.\n");
59. fprintf("===========================\n");
60. diary off;
61. beep;
62. % Podmienka: Ak je prvým vstupným argumentom cesta k priečinku, zavolá sa
63. % funkcia create\_values.:
64. else
65. % Volanie funkcie create\_values.:
66. [values\_clean\_NEW, values\_excluded\_NEW, file\_count] = ...
67. create\_values(OLD\_DATA\_PATH, STEP, UNITS, PROGRESS);
68. end
69. % Následne funkcia s názvom create\_missing vytvorí tabuľku s názvom
70. % missing\_NEW, v ktorej budú uložené všetky údaje o chýbajúcich, resp.
71. % vyradených údajoch.:
72. if exist("PROGRESS", 'var')
73. pd\_state = file\_count + 1;
74. PROGRESS.Value = pd\_state / (file\_count + 13);
75. PROGRESS.Message = "Vytvára sa tabuľka s názvom missing\_NEW...";
76. end
77. missing\_NEW = create\_missing(values\_clean\_NEW, STEP);
78. % Táto tabuľka bude následne rozšírená o niekoľko stĺpcov, v ktorých budú
79. % uložené vyradené údaje a tieto údaje doplnené o chýbajúce hodnoty. Nové
80. % údaje budú uložené v časovej tabuľke s názvom values\_filled\_NEW.:
81. if exist("PROGRESS", 'var')
82. pd\_state = pd\_state + 1;
83. PROGRESS.Value = pd\_state / (file\_count + 13);
84. PROGRESS.Message = "Dopĺňajú sa chýbajúce hodnoty...";
85. end
86. [values\_filled\_NEW, missing\_NEW] = ...
87. create\_filled(missing\_NEW, values\_excluded\_NEW, STEP, TOLERANCE);
88. % Táto časť kódu spája čisté a doplnené hodnoty a následne ich triedi podľa
89. % času.:
90. if exist("PROGRESS", 'var')
91. pd\_state = pd\_state + 1;
92. PROGRESS.Value = pd\_state / (file\_count + 13);
93. PROGRESS.Message = "Zlučujú sa čisté a doplnené hodnoty...";
94. end
95. values\_clean\_NEW = [values\_clean\_NEW; values\_filled\_NEW];
96. values\_clean\_NEW = sortrows(values\_clean\_NEW, "Time");
97. % Táto časť kódu vytvorí tabuľky s názvom stats\_YYYY\_NEW
98. % (pričom YYYY = rok) pre všetky definované časové intervaly uložené v
99. % premennej SIGNAL\_LENGTHS.:
100. SIGNAL\_LENGTHS = [10, 12, 15, 20, 30, 60];
101. SIGNAL\_LENGTHS\_STRINGS\_NEW = strings(length(SIGNAL\_LENGTHS), 1);
102. idx = 0;
103. for i = SIGNAL\_LENGTHS
104. if exist("PROGRESS", 'var')
105. pd\_state = pd\_state + 1;
106. PROGRESS.Value = pd\_state / (file\_count + 13);
107. PROGRESS.Message = ...
108. "Vytvára sa tabuľka štatistických parametrov pre " + i + ...
109. "-minútové intervaly...";
110. end
111. idx = idx + 1;
112. SIGNAL\_LENGTHS\_STRINGS\_NEW(idx) = i + " min";
113. eval("stats\_" + i + "\_NEW = create\_statistics(values\_clean\_NEW, i);");
114. end
115. % Táto časť kódu rozdelí čisté údaje podľa rokov:
116. if exist("PROGRESS", 'var')
117. pd\_state = pd\_state + 1;
118. PROGRESS.Value = pd\_state / (file\_count + 13);
119. PROGRESS.Message = "Rozdeľujú sa údaje podľa rokov...";
120. end
121. YEARS = unique(values\_clean\_NEW.Time.Year);
122. YEARS\_STRINGS\_NEW = string(YEARS);
123. for i = 1:size(YEARS, 1)
124. eval("values\_clean\_" + YEARS(i) + ...
125. "\_NEW = values\_clean\_NEW(values\_clean\_NEW.Time.Year == " + ...
126. "YEARS(i), :);");
127. eval("values\_excluded\_" + YEARS(i) + ...
128. "\_NEW = values\_excluded\_NEW(values\_excluded\_NEW.Time.Year == " + ...
129. "YEARS(i), :);");
130. if ~isempty(values\_filled\_NEW)
131. eval("values\_filled\_" + YEARS(i) + ...
132. "\_NEW = values\_filled\_NEW(values\_filled\_NEW.Time.Year == " + ...
133. "YEARS(i), :);");
134. end
135. for j = SIGNAL\_LENGTHS
136. eval("stats\_" + j + "\_" + YEARS(i) + ...
137. "\_NEW = stats\_" + j + "\_NEW(stats\_" + j + ...
138. "\_NEW.DateStart.Year == YEARS(i), :);");
139. end
140. end
141. % Táto časť otvorí súčasné údaje:
142. if exist("PROGRESS", 'var')
143. pd\_state = pd\_state + 1;
144. PROGRESS.Value = pd\_state / (file\_count + 13);
145. PROGRESS.Message = "Otvárajú sa súčasné údaje...";
146. end
147. NEW\_DATA\_PATH = BASE\_PATH + "\2. Pripravené údaje\";
148. ANALYSIS\_PATH = BASE\_PATH + "\3. Analýza údajov\";
149. MISSING\_PATH = BASE\_PATH + "\3. Analýza údajov\" + ...
150. "Chyby údajov\Automaticky upravené riadky\";
151. for i = 1:size(YEARS, 1)
152. try
153. for j = 1:size(SIGNAL\_LENGTHS, 2)
154. load(NEW\_DATA\_PATH + "stats\_" + YEARS(i) + ...
155. ".mat", "stats\_" + SIGNAL\_LENGTHS(j) + "\_" + YEARS(i));
156. end
157. load(NEW\_DATA\_PATH + "values\_clean\_" + YEARS(i) + ...
158. ".mat", "values\_clean\_" + YEARS(i));
159. load(NEW\_DATA\_PATH + "values\_excluded\_" + YEARS(i) + ...
160. ".mat", "values\_excluded\_" + YEARS(i));
161. if ~isempty(values\_filled\_NEW)
162. load(NEW\_DATA\_PATH + "values\_filled\_" + YEARS(i) + ...
163. ".mat", "values\_filled\_" + YEARS(i));
164. end
165. catch
166. for j = 1:size(SIGNAL\_LENGTHS, 2)
167. eval("stats\_" + SIGNAL\_LENGTHS(j) + "\_" + YEARS(i) + " = [];");
168. end
169. eval("values\_clean\_" + YEARS(i) + " = [];");
170. eval("values\_excluded\_" + YEARS(i) + " = [];");
171. if ~isempty(values\_filled\_NEW)
172. eval("values\_filled\_" + YEARS(i) + " = [];");
173. end
174. end
175. end
176. load(ANALYSIS\_PATH + "years.mat", "YEARS\_STRINGS");
177. load(ANALYSIS\_PATH + "signal\_lengths.mat", "SIGNAL\_LENGTHS\_STRINGS");
178. load(MISSING\_PATH + "missing.mat", "missing");
179. % Táto časť kódu pridá k súčasným údajom nové údaje s koncovkou \_NEW:
180. if exist("PROGRESS", 'var')
181. pd\_state = pd\_state + 1;
182. PROGRESS.Value = pd\_state / (file\_count + 13);
183. PROGRESS.Message = "Pridávajú sa nové údaje...";
184. end
185. for i = 1:size(YEARS, 1)
186. for j = 1:size(SIGNAL\_LENGTHS, 2)
187. eval("stats\_" + SIGNAL\_LENGTHS(j) + "\_" + YEARS(i) + ...
188. " = [stats\_" + SIGNAL\_LENGTHS(j) + "\_" + YEARS(i) + ...
189. "; stats\_" + SIGNAL\_LENGTHS(j) + "\_" + YEARS(i) + "\_NEW];");
190. end
191. eval("values\_clean\_" + YEARS(i) + " = [values\_clean\_" + YEARS(i) + ...
192. "; values\_clean\_" + YEARS(i) + "\_NEW];");
193. eval("values\_excluded\_" + YEARS(i) + " = [values\_excluded\_" + ...
194. YEARS(i) + "; values\_excluded\_" + YEARS(i) + "\_NEW];");
195. if ~isempty(values\_filled\_NEW)
196. eval("values\_filled\_" + YEARS(i) + " = [values\_filled\_" + ...
197. YEARS(i) + "; values\_filled\_" + YEARS(i) + "\_NEW];");
198. end
199. end
200. YEARS\_STRINGS = unique([YEARS\_STRINGS; YEARS\_STRINGS\_NEW]);
201. SIGNAL\_LENGTHS\_STRINGS = ...
202. unique([SIGNAL\_LENGTHS\_STRINGS; SIGNAL\_LENGTHS\_STRINGS\_NEW]);
203. if isempty(missing\_NEW)
204. missing\_NEW = [];
205. elseif size(missing, 2) ~= size(missing\_NEW, 2)
206. missing.NEW\_Data = cell(height(missing), 1);
207. missing.NEW\_Missing = zeros(height(missing), 1);
208. end
209. missing = [missing; missing\_NEW];
210. % Táto časť kódu uloží všetky údaje do príslušných priečinkov:
211. if exist("PROGRESS", 'var')
212. pd\_state = pd\_state + 1;
213. PROGRESS.Value = pd\_state / (file\_count + 13);
214. PROGRESS.Message = "Ukladajú sa údaje...";
215. end
216. for i = 1:size(YEARS, 1)
217. save(NEW\_DATA\_PATH + "stats\_" + YEARS(i) + ".mat", "stats\_" + ...
218. SIGNAL\_LENGTHS(1) + "\_" + YEARS(i), "-v7.3");
219. for j = 2:length(SIGNAL\_LENGTHS)
220. save(NEW\_DATA\_PATH + "stats\_" + YEARS(i) + ".mat", "stats\_" + ...
221. SIGNAL\_LENGTHS(j) + "\_" + YEARS(i), "-append");
222. end
223. save(NEW\_DATA\_PATH + "values\_clean\_" + YEARS(i) + ".mat", ...
224. "values\_clean\_" + YEARS(i), "-v7.3");
225. save(NEW\_DATA\_PATH + "values\_excluded\_" + YEARS(i) + ".mat", ...
226. "values\_excluded\_" + YEARS(i), "-v7.3");
227. if ~isempty(values\_filled\_NEW)
228. save(NEW\_DATA\_PATH + "values\_filled\_" + YEARS(i) + ".mat", ...
229. "values\_filled\_" + YEARS(i), "-v7.3");
230. end
231. end
232. save(ANALYSIS\_PATH + "years.mat", "YEARS\_STRINGS", "-v7.3");
233. save(ANALYSIS\_PATH + "signal\_lengths.mat", "SIGNAL\_LENGTHS\_STRINGS", "-v7.3");
234. save(MISSING\_PATH + "missing.mat", "missing", "-v7.3");
235. end

**8. Kód aplikácie *ETL\_Configuration***

A.) Funkcie spätných volaní

1. % Kód, ktorý sa vykoná hneď po otvorení aplikácie.:
2. function startupFcn(app)
3. % Nastaví aplikácii šírku (W) a výšku (H) v pixeloch a umiestni
4. % ju na stred obrazovky.:
5. W = 640; H = 255;
6. R = get(0, 'screensize');
7. app.UIFigure.Position = [(R(3) / 2) - (W / 2), (R(4) / 2) - (H / 2), W, H];
8. uiprogressdlg(app.UIFigure, 'Title', "Počkajte prosím...", "Indeterminate", "on");
9. end
10. % Funkcia stlačenia tlačidla: Tlacidlo\_Dalej
11. function Tlacidlo\_DalejButtonPushed(app, event)
12. if app.Tlacidlo\_Dalej.Text == "Spustiť"
13. % Otvorí dialógové okno progresu:
14. pd = uiprogressdlg(app.UIFigure, 'Title', "Počkajte prosím...", ...
15. "Message", "Spúšťa sa výpočet...");
16. % Táto časť kódu vyberie z textových polí hodnoty a uloží
17. % do nasledujúcich premenných:
18. cesta = string(app.Text\_Vlozte\_Cestu.Value);
19. krok = app.Numeric\_Krokovanie.Value;
20. jednotka = string(app.Text\_Jednotky.Value);
21. tolerancia = app.Numeric\_Tolerancia.Value;
22. % Táto časť kódu v závislosti od výberu operácie spustí operáciu:
23. if app.List\_Vyberte\_Operaciu.Value == "ETL Update"
24. etl\_update(cesta, krok, jednotka,tolerancia, pd);
25. elseif app.List\_Vyberte\_Operaciu.Value == "ETL Append"
26. etl\_append(cesta, krok, jednotka,tolerancia, pd);
27. end
28. % Zatvorí dialógové okno progresu:
29. pd.Title = "ETL operácia bola úspešne dokončená.";
30. pd.Value = 1;
31. pd.Message = ""
32. beep;
33. pause(5);
34. close(pd);
35. app.Tlacidlo\_Dalej.Enable = false;
36. elseif app.Ziarovka\_5.Visible == "on"
37. % Táto časť kódu prepíše popis nasledujúceho kroku:
38. app.Title.Text = "Krok č. 6 : Skontrolujte nastavenia pred spustením operácie.";
39. % Zmení tlačidlo "Ďalej" na "Spustiť":
40. app.Tlacidlo\_Dalej.Text = "Spustiť";
41. % Zmení farbu žiarovky na zelenú:
42. app.Ziarovka\_5.Color = [0, 1, 0];
43. % Zablokuje úpravu tohto kroku:
44. app.Label\_Tolerancia.Enable = false;
45. app.Numeric\_Tolerancia.Enable = false;
46. elseif app.Ziarovka\_4.Visible == "on"
47. % Táto časť kódu prepíše v závislosti od výberu operácie popis nasledujúceho kroku:
48. if app.List\_Vyberte\_Operaciu.Value == "ETL Update"
49. app.Title.Text = "Krok č. 5 : Nastavte hornú hranicu počtu hodín (vrátane), ktoré majú byť doplnené.";
50. elseif app.List\_Vyberte\_Operaciu.Value == "ETL Append"
51. app.Title.Text = "Krok č. 5 : .";
52. end
53. % Zmení farbu žiarovky na zelenú a odkryje nasledujúcu žiarovku:
54. app.Ziarovka\_4.Color = [0, 1, 0];
55. app.Ziarovka\_5.Visible = "on";
56. % Zablokuje úpravu tohto kroku a odkryje nasledujúci krok:
57. app.Label\_Jednotky.Enable = false;
58. app.Text\_Jednotky.Enable = false;
59. app.Label\_Tolerancia.Visible = true;
60. app.Numeric\_Tolerancia.Visible = true;
61. elseif app.Ziarovka\_3.Visible == "on"
62. % Táto časť kódu prepíše v závislosti od výberu operácie popis nasledujúceho kroku:
63. if app.List\_Vyberte\_Operaciu.Value == "ETL Update"
64. app.Title.Text = "Krok č. 4 : Ak je v údajoch aj jednotka meranej veličiny, uveďte ju. Ak nie, nechajte toto pole prázdne.";
65. elseif app.List\_Vyberte\_Operaciu.Value == "ETL Append"
66. app.Title.Text = "Krok č. 4 : .";
67. end
68. % Zmení farbu žiarovky na zelenú a odkryje nasledujúcu žiarovku:
69. app.Ziarovka\_3.Color = [0, 1, 0];
70. app.Ziarovka\_4.Visible = "on";
71. % Zablokuje úpravu tohto kroku a odkryje nasledujúci krok:
72. app.Label\_Krokovanie.Enable = false;
73. app.Numeric\_Krokovanie.Enable = false;
74. app.Label\_Jednotky.Visible = true;
75. app.Text\_Jednotky.Visible = true;
76. elseif app.Ziarovka\_2.Visible == "on"
77. % Táto časť kódu prepíše v závislosti od výberu operácie popis nasledujúceho kroku:
78. if app.List\_Vyberte\_Operaciu.Value == "ETL Update"
79. app.Title.Text = "Krok č. 3 : Uveďte veľkosť krokov merania v sekundách.";
80. elseif app.List\_Vyberte\_Operaciu.Value == "ETL Append"
81. app.Title.Text = "Krok č. 3 : .";
82. end
83. % Zmení farbu žiarovky na zelenú a odkryje nasledujúcu žiarovku:
84. app.Ziarovka\_2.Color = [0, 1, 0];
85. app.Ziarovka\_3.Visible = "on";
86. % Zablokuje úpravu tohto kroku a odkryje nasledujúci krok:
87. app.Label\_Vlozte\_Cestu.Enable = false;
88. app.Text\_Vlozte\_Cestu.Enable = false;
89. app.Label\_Krokovanie.Visible = true;
90. app.Numeric\_Krokovanie.Visible = true;
91. elseif app.Ziarovka\_1.Visible == "on"
92. % Táto časť kódu prepíše v závislosti od výberu operácie popis nasledujúceho kroku:
93. if app.List\_Vyberte\_Operaciu.Value == "ETL Update"
94. app.Title.Text = "Krok č. 2 : Vložte cestu k priečinku s \*.CSV súbormi.";
95. elseif app.List\_Vyberte\_Operaciu.Value == "ETL Append"
96. app.Title.Text = "Krok č. 2 : Vložte cestu k priečinku s \*.CSV súbormi alebo cestu k \*.CSV súboru.";
97. end
98. % Zmení farbu žiarovky na zelenú a odkryje nasledujúcu žiarovku:
99. app.Ziarovka\_1.Color = [0, 1, 0];
100. app.Ziarovka\_2.Visible = "on";
101. % Povolí možnosť zatlačenia tlačidla Späť:
102. app.Tlacidlo\_Spat.Enable = true;
103. % Zablokuje úpravu tohto kroku a odkryje nasledujúci krok:
104. app.Label\_Vyberte\_Operaciu.Enable = false;
105. app.List\_Vyberte\_Operaciu.Enable = false;
106. app.Label\_Vlozte\_Cestu.Visible = true;
107. app.Text\_Vlozte\_Cestu.Visible = true;
108. end
109. end
110. % Funkcia stlačenia tlačidla: Tlacidlo\_Spat
111. function Tlacidlo\_SpatButtonPushed(app, event)
112. if app.Tlacidlo\_Dalej.Text == "Spustiť"
113. % Táto časť kódu prepíše v závislosti od výberu operácie popis predchádzajúceho kroku:
114. if app.List\_Vyberte\_Operaciu.Value == "ETL Update"
115. app.Title.Text = "Krok č. 5 : Nastavte hornú hranicu počtu hodín (vrátane), ktoré majú byť doplnené.";
116. elseif app.List\_Vyberte\_Operaciu.Value == "ETL Append"
117. app.Title.Text = "Krok č. 5 : .";
118. end
119. % Zmení tlačidlo "Spustiť" na "Ďalej":
120. app.Tlacidlo\_Dalej.Text = "Ďalej";
121. app.Tlacidlo\_Dalej.Enable = true;
122. % Zmení farbu žiarovky na sivú:
123. app.Ziarovka\_5.Color = [0.9, 0.9, 0.9];
124. % Odblokuje úpravu predchádzajúceho kroku:
125. app.Label\_Tolerancia.Enable = true;
126. app.Numeric\_Tolerancia.Enable = true;
127. elseif app.Ziarovka\_5.Visible == "on"
128. % Táto časť kódu prepíše v závislosti od výberu operácie popis predchádzajúceho kroku:
129. if app.List\_Vyberte\_Operaciu.Value == "ETL Update"
130. app.Title.Text = "Krok č. 4 : Ak je v údajoch aj jednotka meranej veličiny, uveďte ju. Ak nie, nechajte toto pole prázdne.";
131. elseif app.List\_Vyberte\_Operaciu.Value == "ETL Append"
132. app.Title.Text = "Krok č. 4 : .";
133. end
134. % Zmení farbu predchádzajúcej žiarovky na sivú a zakryje túto žiarovku:
135. app.Ziarovka\_4.Color = [0.9, 0.9, 0.9];
136. app.Ziarovka\_5.Visible = "off";
137. % Odblokuje úpravu predchádzajúceho kroku a zakryje tento krok:
138. app.Label\_Jednotky.Enable = true;
139. app.Text\_Jednotky.Enable = true;
140. app.Label\_Tolerancia.Visible = false;
141. app.Numeric\_Tolerancia.Visible = false;
142. elseif app.Ziarovka\_4.Visible == "on"
143. % Táto časť kódu prepíše v závislosti od výberu operácie popis predchádzajúceho kroku:
144. if app.List\_Vyberte\_Operaciu.Value == "ETL Update"
145. app.Title.Text = "Krok č. 3 : Uveďte veľkosť krokov merania v sekundách.";
146. elseif app.List\_Vyberte\_Operaciu.Value == "ETL Append"
147. app.Title.Text = "Krok č. 3 : .";
148. end
149. % Zmení farbu predchádzajúcej žiarovky na sivú a zakryje túto žiarovku:
150. app.Ziarovka\_3.Color = [0.9, 0.9, 0.9];
151. app.Ziarovka\_4.Visible = "off";
152. % Odblokuje úpravu predchádzajúceho kroku a zakryje tento krok:
153. app.Label\_Krokovanie.Enable = true;
154. app.Numeric\_Krokovanie.Enable = true;
155. app.Label\_Jednotky.Visible = false;
156. app.Text\_Jednotky.Visible = false;
157. elseif app.Ziarovka\_3.Visible == "on"
158. % Táto časť kódu prepíše v závislosti od výberu operácie popis predchádzajúceho kroku:
159. if app.List\_Vyberte\_Operaciu.Value == "ETL Update"
160. app.Title.Text = "Krok č. 2 : Vložte cestu k priečinku s \*.CSV súbormi.";
161. elseif app.List\_Vyberte\_Operaciu.Value == "ETL Append"
162. app.Title.Text = "Krok č. 2 : Vložte cestu k priečinku s \*.CSV súbormi alebo cestu k \*.CSV súboru.";
163. end
164. % Zmení farbu predchádzajúcej žiarovky na sivú a zakryje túto žiarovku:
165. app.Ziarovka\_2.Color = [0.9, 0.9, 0.9];
166. app.Ziarovka\_3.Visible = "off";
167. % Odblokuje úpravu predchádzajúceho kroku a zakryje tento krok:
168. app.Label\_Vlozte\_Cestu.Enable = true;
169. app.Text\_Vlozte\_Cestu.Enable = true;
170. app.Label\_Krokovanie.Visible = false;
171. app.Numeric\_Krokovanie.Visible = false;
172. elseif app.Ziarovka\_2.Visible == "on"
173. % Táto časť kódu prepíše popis predchádzajúceho kroku:
174. app.Title.Text = "Krok č. 1 : Vyberte operáciu automatizovaného ETL procesu.";
175. app.Tlacidlo\_Spat.Enable = false;
176. % Zmení farbu predchádzajúcej žiarovky na sivú a zakryje túto žiarovku:
177. app.Ziarovka\_1.Color = [0.9, 0.9, 0.9];
178. app.Ziarovka\_2.Visible = "off";
179. % Odblokuje úpravu predchádzajúceho kroku a zakryje tento krok:
180. app.Label\_Vyberte\_Operaciu.Enable = true;
181. app.List\_Vyberte\_Operaciu.Enable = true;
182. app.Label\_Vlozte\_Cestu.Visible = false;
183. app.Text\_Vlozte\_Cestu.Visible = false;
184. end
185. end
186. % Došlo k zmene hodnoty: List\_Vyberte\_Operaciu
187. function List\_Vyberte\_OperaciuValueChanged(app, event)
188. % Ak je v liste možnosť "Vybrať", tak túto možnosť odstráni:
189. if app.List\_Vyberte\_Operaciu.Items(1) == "Vybrať"
190. app.List\_Vyberte\_Operaciu.Items(1) = [];
191. end
193. % Povolí možnosť zatlačenia tlačidla Ďalej:
194. app.Tlacidlo\_Dalej.Enable = true;
195. end
196. % Funkcia stlačenia tlačidla: Tlacidlo\_Zrusit
197. function Tlacidlo\_ZrusitButtonPushed(app, event)
198. % Zatvorí aplikáciu.:
199. closereq;
200. end

B.) Pomocné funkcie

Aplikácia *ETL\_Configuration* nemá žiadne pomocné funkcie.

C.) Vlastnosti aplikácie

Aplikácia *ETL\_Configuration* nemá žiadne vlastnosti.

**9.** **Kód aplikácie *Descriptive\_Statistic***

A.) Funkcie spätných volaní

1. % Kód, ktorý sa vykoná hneď po otvorení aplikácie.:
2. function startupFcn(app, year)
3. % Nastaví aplikácii šírku (W) a výšku (H) v pixeloch a umiestni ju na stred obrazovky.:
4. W = 1920; H = 1080;
5. R = get(0, 'screensize');
6. app.DescriptiveStatistic.Position = [(R(3) / 2) - (W / 2), (R(4) / 2) - (H / 2), W, H];
7. uiprogressdlg(app.DescriptiveStatistic, 'Title', "Please Wait", "Indeterminate", "on");
8. % Zakryje všetky objekty v aplikácii na oboch záložkach a ponechá len prvý objekt.:
9. ds\_objects\_visibility(app, false);
10. pos\_objects\_visibility(app, false);
11. % Načíta zo súboru years.mat list rokov a nahradí rozbaľovacie listy týmito hodnotami
12. % na oboch záložkach.:
13. app.Years = load("years.mat", "YEARS\_STRINGS");
14. app.Years = app.Years.YEARS\_STRINGS;
15. app.YearDropDown.Items = ["Select"; app.Years];
16. app.YearDropDown\_2.Items = ["Select"; app.Years];
17. % Načíta zo súboru signal\_lengths.mat list dĺžok signálov a nahradí rozbaľovacie listy
18. % týmito hodnotami na oboch záložkach.:
19. app.Signal\_Lengths = load("signal\_lengths.mat", "SIGNAL\_LENGTHS\_STRINGS");
20. app.Signal\_Lengths = app.Signal\_Lengths.SIGNAL\_LENGTHS\_STRINGS;
21. app.SignalLengthDropDown.Items = ["Select"; app.Signal\_Lengths];
22. app.SignalLengthDropDown\_2.Items = ["Select"; app.Signal\_Lengths];
23. % Vygeneruje paletu konkrétných RGB farieb pre krivky grafov vykreslené na oboch
24. % záložkach.:
25. app.Palette = [...
26. 0.0000, 0.0000, 1.0000; ...
27. 1.0000, 0.0000, 0.0000; ...
28. 0.0431, 0.4000, 0.1372; ...
29. 0.0000, 0.0000, 0.0000; ...
30. 0.5882, 0.2941, 0.0000; ...
31. ];
32. end
33. % Došlo k zmene hodnoty: SignalLengthDropDown
34. function SignalLengthDropDownValueChanged(app, event)
35. uiprogressdlg(app.DescriptiveStatistic, 'Title', "Please Wait", "Indeterminate", "on");
36. % Odstráni možnosť "Select" zo zoznamu pri prvom výbere.:
37. if app.SignalLengthDropDown.Items(1) == "Select"
38. app.SignalLengthDropDown.Items(1) = [];
39. else
40. app.YearDropDown.Items = ["Select"; app.Years];
41. app.YearDropDown.Value = app.YearDropDown.Items(1);
42. ds\_gui\_visibility(app, false);
43. end
44. % Zobrazí objekty SignalLengthDropDown a YearDropDown.:
45. ds\_objects\_visibility(app, false);
46. app.SignalLengthDropDown.Visible = true;
47. app.SignalLengthDropDown.Enable = true;
48. app.SignalLengthLabel.Visible = true;
49. app.SignalLengthLabel.Enable = true;
50. app.YearDropDown.Enable = true;
51. app.YearDropDown.Visible = true;
52. app.YearDropDownLabel.Enable = true;
53. app.YearDropDownLabel.Visible = true;
54. end
55. % Došlo k zmene hodnoty: SignalLengthDropDown\_2
56. function SignalLengthDropDown\_2ValueChanged(app, event)
57. uiprogressdlg(app.DescriptiveStatistic, 'Title', "Please Wait", "Indeterminate", "on");
58. % Odstráni možnosť "Select" zo zoznamu pri prvom výbere.:
59. if app.SignalLengthDropDown\_2.Items(1) == "Select"
60. app.SignalLengthDropDown\_2.Items(1) = [];
61. else
62. app.YearDropDown\_2.Items = ["Select"; app.Years];
63. app.YearDropDown\_2.Value = app.YearDropDown\_2.Items(1);
64. pos\_gui\_visibility(app, false);
65. end
66. % Zobrazí objekty SignalLengthDropDown\_2 a YearDropDown\_2.:
67. pos\_objects\_visibility(app, false);
68. app.SignalLengthDropDown\_2.Visible = true;
69. app.SignalLengthDropDown\_2.Enable = true;
70. app.SignalLengthLabel\_2.Visible = true;
71. app.SignalLengthLabel\_2.Enable = true;
72. app.YearDropDown\_2.Enable = true;
73. app.YearDropDown\_2.Visible = true;
74. app.YearDropDown\_2Label.Enable = true;
75. app.YearDropDown\_2Label.Visible = true;
76. end
77. % Došlo k zmene hodnoty: YearDropDown
78. function YearDropDownValueChanged(app, event)
79. uiprogressdlg(app.DescriptiveStatistic, 'Title', "Please Wait", "Indeterminate", "on");
80. app.CurrentYear = app.YearDropDown.Value;
81. % Načíta časovú tabuľku čistých údajov a uloží ju do vlastnosti aplikácie Values\_1 pre
82. % prvú záložku.:
83. app.Values\_1 = load("values\_clean\_" + app.CurrentYear + ".mat", "values\_clean\_" + app.CurrentYear);
84. eval("app.Values\_1 = app.Values\_1.values\_clean\_" + app.CurrentYear + ";");
85. if app.YearDropDown.Items(1) == "Select"
86. app.YearDropDown.Items(1) = [];
87. else
88. app.DateDropDown.Items = ["Select", app.DateDropDown.Items];
89. app.DateDropDown.Value = app.DateDropDown.Items(1);
90. ds\_gui\_visibility(app, false);
91. end
92. % Načíta tabuľku vypočítaných štatistických parametrov a uloží ju do vlastnosti
93. % aplikácie Stats\_1 pre prvú záložku.:
94. MINUTES = erase(app.SignalLengthDropDown.Value, " min");
95. MINUTES = str2double(MINUTES);
96. app.Interval\_Length = seconds(minutes(MINUTES));
97. app.Stats\_1 = load("stats\_" + app.CurrentYear + ".mat", "stats\_" + MINUTES + "\_" + app.CurrentYear);
98. eval("app.Stats\_1 = app.Stats\_1.stats\_" + MINUTES + "\_" + app.CurrentYear + ";");
99. % Zobrazí objekty SignalLengthDropDown, YearDropDown a DateDropDown.:
100. ds\_objects\_visibility(app, false);
101. app.SignalLengthDropDown.Visible = true;
102. app.SignalLengthDropDown.Enable = true;
103. app.SignalLengthLabel.Visible = true;
104. app.SignalLengthLabel.Enable = true;
105. app.YearDropDown.Enable = true;
106. app.YearDropDown.Visible = true;
107. app.YearDropDownLabel.Enable = true;
108. app.YearDropDownLabel.Visible = true;
109. app.DateDropDown.Enable = true;
110. app.DateDropDown.Visible = true;
111. app.DateDropDownLabel.Enable = true;
112. app.DateDropDownLabel.Visible = true;
113. % Načíta list dostupných dátumov dní do rozbaľovacieho listu.:
114. app.DateDropDown.Items = ds\_load\_dates(app);
115. % Nastaví prvú hodnotu z listu dátumov, t.j. hodnota "Select".:
116. app.DateDropDown.Value = app.DateDropDown.Items(1);
117. end
118. % Došlo k zmene hodnoty: YearDropDown\_2
119. function YearDropDown\_2ValueChanged(app, event)
120. uiprogressdlg(app.DescriptiveStatistic, 'Title', "Please Wait", "Indeterminate", "on");
121. app.CurrentYear = app.YearDropDown\_2.Value;
122. % Načíta časovú tabuľku čistých údajov a uloží ju do vlastnosti aplikácie Values\_2 pre
123. % druhú záložku.:
124. app.Values\_2 = load("values\_clean\_" + app.CurrentYear + ".mat", "values\_clean\_" + app.CurrentYear);
125. eval("app.Values\_2 = app.Values\_2.values\_clean\_" + app.CurrentYear + ";");
126. if app.YearDropDown\_2.Items(1) == "Select"
127. app.YearDropDown\_2.Items(1) = [];
128. else
129. app.DateDropDown\_2.Items = ["Select", app.DateDropDown\_2.Items];
130. app.DateDropDown\_2.Value = app.DateDropDown\_2.Items(1);
131. pos\_gui\_visibility(app, false);
132. end
133. % Zobrazí objekty SignalLengthDropDown\_2, YearDropDown\_2 a DateDropDown\_2.:
134. pos\_objects\_visibility(app, false);
135. app.SignalLengthDropDown\_2.Visible = true;
136. app.SignalLengthDropDown\_2.Enable = true;
137. app.SignalLengthLabel\_2.Visible = true;
138. app.SignalLengthLabel\_2.Enable = true;
139. app.YearDropDown\_2.Enable = true;
140. app.YearDropDown\_2.Visible = true;
141. app.YearDropDown\_2Label.Enable = true;
142. app.YearDropDown\_2Label.Visible = true;
143. app.DateDropDown\_2.Enable = true;
144. app.DateDropDown\_2.Visible = true;
145. app.DateDropDown\_2Label.Enable = true;
146. app.DateDropDown\_2Label.Visible = true;
147. % Načíta tabuľku vypočítaných štatistických parametrov a uloží ju do vlastnosti
148. % aplikácie Stats\_2 pre druhú záložku.:
149. MINUTES = erase(app.SignalLengthDropDown\_2.Value, " min");
150. MINUTES = str2double(MINUTES);
151. app.Interval\_Length = seconds(minutes(MINUTES));
152. app.Stats\_2 = load("stats\_" + app.CurrentYear + ".mat", "stats\_" + MINUTES + "\_" + app.CurrentYear);
153. eval("app.Stats\_2 = app.Stats\_2.stats\_" + MINUTES + "\_" + app.CurrentYear + ";");
154. % Načíta list dostupných dátumov dní do rozbaľovacieho listu.:
155. app.DateDropDown\_2.Items = pos\_load\_dates(app);
156. % Nastaví prvú hodnotu z listu dátumov, t.j. hodnota "Select".:
157. app.DateDropDown\_2.Value = app.DateDropDown\_2.Items(1);
158. end
159. % Došlo k zmene hodnoty: DateDropDown
160. function DateDropDownValueChanged(app, event)
161. uiprogressdlg(app.DescriptiveStatistic, 'Title', "Please Wait", "Indeterminate", "on");
162. % Táto časť sa spustí raz na začiatku (keď bol dátum vybraný prvýkrát).:
163. if app.DateDropDown.Items(1) == "Select"
164. app.DateDropDown.Items(1) = [];
165. ds\_objects\_visibility(app, true);
166. End
167. % Zavolá sa najprv pomocná funkcia ds\_find\_index na vyhľadanie indexu v závislosti od
168. % zvoleného dňa. Následne sa zavolá hlavná funkcia ds\_update na aktualizáciu zobrazenia
169. % prvej záložky.:
170. Index = ds\_find\_index(app, app.DateDropDown);
171. ds\_update(app, Index(1), Index(2));
172. end
173. % Došlo k zmene hodnoty: DateDropDown\_2
174. function DateDropDown\_2ValueChanged(app, event)
175. uiprogressdlg(app.DescriptiveStatistic, 'Title', "Please Wait", "Indeterminate", "on");
176. % Táto časť sa spustí raz na začiatku (keď bol dátum vybraný prvýkrát) .:
177. if app.DateDropDown\_2.Items(1) == "Select"
178. app.DateDropDown\_2.Items(1) = [];
179. pos\_objects\_visibility(app, true);
180. end
181. % Zavolá sa najprv pomocná funkcia pos\_find\_index na vyhľadanie indexu v závislosti od
182. % zvoleného dňa. Následne sa zavolá hlavná funkcia pos\_update na aktualizáciu
183. % zobrazenia druhej záložky.:
184. Index = pos\_find\_index(app, app.DateDropDown\_2);
185. pos\_update(app, Index(1), Index(2));
186. end
187. % Funkcia stlačenia tlačidla: PreviousButton
188. function PreviousButtonPushed(app, event)
189. uiprogressdlg(app.DescriptiveStatistic, 'Title', "Please Wait", "Indeterminate", "on");
190. % Funkcia uloží pozície aktuálneho výberu všetkých rozbaľovacích listov.:
191. idx\_signal\_length = find(ismember(app.SignalLengthDropDown.Items, app.SignalLengthDropDown.Value), 1);
192. idx\_year = find(ismember(app.YearDropDown.Items, app.YearDropDown.Value), 1);
193. idx\_date = find(ismember(app.DateDropDown.Items, app.DateDropDown.Value), 1);
194. % Pomocná logická premenná, ktorá overí či je aktuálne zvolený prvý deň v roku.:
195. is\_date\_first = idx\_date == size(app.DateDropDown.Items, 2);
196. % Ak je zvolený prvý deň v roku, zobrazenie sa presunie automaticky do
197. % prechadzajúceho roku posledného dostupného dňa.:
198. if is\_date\_first
199. app.SignalLengthDropDown.Value = app.SignalLengthDropDown.Items(idx\_signal\_length);
200. SignalLengthDropDownValueChanged(app);
201. app.YearDropDown.Value = app.YearDropDown.Items(idx\_year);
202. YearDropDownValueChanged(app);
203. app.DateDropDown.Value = app.DateDropDown.Items(1);
204. DateDropDownValueChanged(app);
205. % V opačnom prípade sa zobrazenie presunie na predchadzajúci deň.:
206. else
207. app.DateDropDown.Value = app.DateDropDown.Items(idx\_date + 1);
208. DateDropDownValueChanged(app);
209. end
210. end
211. % Funkcia stlačenia tlačidla: PreviousButton\_2
212. function PreviousButton\_2Pushed(app, event)
213. uiprogressdlg(app.DescriptiveStatistic, 'Title', "Please Wait", "Indeterminate", "on");
214. % Funkcia uloží pozície aktuálneho výberu všetkých rozbaľovacích listov.:
215. idx\_signal\_length = find(ismember(app.SignalLengthDropDown\_2.Items, app.SignalLengthDropDown\_2.Value), 1);
216. idx\_year = find(ismember(app.YearDropDown\_2.Items, app.YearDropDown\_2.Value), 1);
217. idx\_date = find(ismember(app.DateDropDown\_2.Items, app.DateDropDown\_2.Value), 1);
218. % Pomocná logická premenná, ktorá overí či je aktuálne zvolený prvý deň v roku.:
219. is\_date\_first = idx\_date == size(app.DateDropDown\_2.Items, 2);
220. % Ak je zvolený prvý deň v roku, zobrazenie sa presunie automaticky do
221. % prechadzajúceho roku posledného dostupného dňa.:
222. if is\_date\_first
223. app.SignalLengthDropDown\_2.Value = app.SignalLengthDropDown\_2.Items(idx\_signal\_length);
224. SignalLengthDropDown\_2ValueChanged(app);
225. app.YearDropDown\_2.Value = app.YearDropDown\_2.Items(idx\_year);
226. YearDropDown\_2ValueChanged(app);
227. app.DateDropDown\_2.Value = app.DateDropDown\_2.Items(1);
228. DateDropDown\_2ValueChanged(app);
229. % V opačnom prípade sa zobrazenie presunie na predchadzajúci deň.:
230. else
231. app.DateDropDown\_2.Value = app.DateDropDown\_2.Items(idx\_date + 1);
232. DateDropDown\_2ValueChanged(app);
233. end
234. end
235. % Funkcia stlačenia tlačidla: NextButton
236. function NextButtonPushed(app, event)
237. uiprogressdlg(app.DescriptiveStatistic, 'Title', "Please Wait", "Indeterminate", "on");
238. % Funkcia uloží pozície aktuálneho výberu všetkých rozbaľovacích listov.:
239. idx\_signal\_length = find(ismember(app.SignalLengthDropDown.Items, app.SignalLengthDropDown.Value), 1);
240. idx\_year = find(ismember(app.YearDropDown.Items, app.YearDropDown.Value), 1);
241. idx\_date = find(ismember(app.DateDropDown.Items, app.DateDropDown.Value), 1);
242. % Pomocná logická premenná, ktorá overí či je aktuálne zvolený posledný deň v roku.:
243. is\_date\_last = idx\_date == 1;
244. % Ak je zvolený posledný deň v roku, zobrazenie sa presunie automaticky do
245. % nasledujúceho roku prvého dostupného dňa.:
246. if is\_date\_last
247. app.SignalLengthDropDown.Value = app.SignalLengthDropDown.Items(idx\_signal\_length);
248. SignalLengthDropDownValueChanged(app);
249. app.YearDropDown.Value = app.YearDropDown.Items(idx\_year + 2);
250. YearDropDownValueChanged(app);
251. app.DateDropDown.Value = app.DateDropDown.Items(end);
252. DateDropDownValueChanged(app);
253. % V opačnom prípade sa zobrazenie presunie na nasledujúci deň.:
254. else
255. app.DateDropDown.Value = app.DateDropDown.Items(idx\_date - 1);
256. DateDropDownValueChanged(app);
257. end
258. end
259. % Funkcia stlačenia tlačidla: NextButton\_2
260. function NextButton\_2Pushed(app, event)
261. uiprogressdlg(app.DescriptiveStatistic, 'Title', "Please Wait", "Indeterminate", "on");
262. % Funkcia uloží pozície aktuálneho výberu všetkých rozbaľovacích listov.:
263. idx\_signal\_length = find(ismember(app.SignalLengthDropDown\_2.Items, app.SignalLengthDropDown\_2.Value), 1);
264. idx\_year = find(ismember(app.YearDropDown\_2.Items, app.YearDropDown\_2.Value), 1);
265. idx\_date = find(ismember(app.DateDropDown\_2.Items, app.DateDropDown\_2.Value), 1);
266. % Pomocná logická premenná, ktorá overí či je aktuálne zvolený posledný deň v roku.:
267. is\_date\_last = idx\_date == 1;
268. % Ak je zvolený posledný deň v roku, zobrazenie sa presunie automaticky do
269. % nasledujúceho roku prvého dostupného dňa.:
270. if is\_date\_last
271. app.SignalLengthDropDown\_2.Value = app.SignalLengthDropDown\_2.Items(idx\_signal\_length);
272. SignalLengthDropDown\_2ValueChanged(app);
273. app.YearDropDown\_2.Value = app.YearDropDown\_2.Items(idx\_year + 2);
274. YearDropDown\_2ValueChanged(app);
275. app.DateDropDown\_2.Value = app.DateDropDown\_2.Items(end);
276. DateDropDown\_2ValueChanged(app);
277. % V opačnom prípade sa zobrazenie presunie na nasledujúci deň.:
278. else
279. app.DateDropDown\_2.Value = app.DateDropDown\_2.Items(idx\_date - 1);
280. DateDropDown\_2ValueChanged(app);
281. end
282. end
283. % Došlo k zmene hodnoty: Switch
284. function SwitchValueChanged(app, event)
285. uiprogressdlg(app.DescriptiveStatistic, 'Title', "Please Wait", "Indeterminate", "on");
286. % Táto časť kódu sa aktivuje pri zmene stavu prepínača na druhej záložke. V závislosti
287. % od pozície, do ktorej bol prepínač prepnutý, aktivujú, resp. deaktivujú sa legendy
288. % pre príslušné grafy.:
289. value = app.Switch.Value;
290. if value == "On"
291. legend(app.UIAxes\_2, ["Maximum", "Mean", "Minimum"], 'Location', 'best');
292. legend(app.UIAxes\_3, ["Mode", "Median", "Mean"], 'Location', 'best');
293. legend(app.UIAxes\_5, ["Variance", "SD", "IQR"], 'Location', 'best');
294. legend(app.UIAxes\_6, ["Range", "IQR"], 'Location', 'best');
295. legend(app.UIAxes\_7, ["Variance", "RSD"], 'Location', 'best');
296. legend(app.UIAxes\_8, "Skewness", 'Location', 'best');
297. legend(app.UIAxes\_9, ["Skewness", "Kurtosis"], 'Location', 'best');
298. legend(app.UIAxes\_10, "Kurtosis", 'Location', 'best');
299. elseif value == "Off"
300. for i = [2, 3, 5, 6, 7, 8, 9, 10]
301. eval("legend(app.UIAxes\_" + i + ", 'hide');");
302. end
303. end
304. end
305. % Došlo k zmene hodnoty: Switch\_2
306. function Switch\_2ValueChanged(app, event)
307. uiprogressdlg(app.DescriptiveStatistic, 'Title', "Please Wait", "Indeterminate", "on");
308. % Táto časť kódu sa aktivuje pri zmene stavu prepínača na prvej záložke. V závislosti
309. % od pozície, do ktorej bol prepínač prepnutý, aktivujú, resp. deaktivujú sa legendy
310. % pre príslušné grafy.:
311. value = app.Switch\_2.Value;
312. if value == "On"
313. legend(app.UIAxes\_12, ["Range", "SD"], 'Location', 'best');
314. legend(app.UIAxes\_13, ["Range", "Skewness"], 'Location', 'best');
315. legend(app.UIAxes\_14, ["Range", "Kurtosis"], 'Location', 'best');
316. legend(app.UIAxes\_15, ["Range", "Accuracy Rate"], 'Location', 'best');
317. legend(app.UIAxes\_16, ["SD", "Skewness"], 'Location', 'best');
318. legend(app.UIAxes\_17, ["Kurtosis", "Accuracy Rate"], 'Location', 'best');
319. legend(app.UIAxes\_18, ["SD", "Accuracy Rate"], 'Location', 'best');
320. legend(app.UIAxes\_19, ["Skewness", "Accuracy Rate"], 'Location', 'best');
321. legend(app.UIAxes\_20, ["SD", "Kurtosis"], 'Location', 'best');
322. elseif value == "Off"
323. for i = 12:20
324. eval("legend(app.UIAxes\_" + i + ", 'hide');");
325. end
326. end
327. end
328. % Došlo k zmene hodnoty: UnitsDropDown
329. function UnitsDropDownValueChanged(app, event)
330. uiprogressdlg(app.DescriptiveStatistic, 'Title', "Please Wait", "Indeterminate", "on");
331. % Pokiaľ používateľ zmení jednotky z listu jednotiek, zmenia sa jednotky na všetkých
332. % príslušných grafoch prvej záložky zobrazenia.:
333. for i = 1:7
334. eval("app.UIAxes\_" + i + ".YLabel.String = app.UnitsDropDown.Value;");
335. end
336. end
337. % Došlo k zmene hodnoty: UnitsDropDown\_2
338. function UnitsDropDown\_2ValueChanged(app, event)
339. uiprogressdlg(app.DescriptiveStatistic, 'Title', "Please Wait", "Indeterminate", "on");
340. % Pokiaľ používateľ zmení jednotky z listu jednotiek, zmenia sa jednotky na všetkých
341. % príslušných grafoch druhej záložky zobrazenia.:
342. for i = 11:20
343. eval("app.UIAxes\_" + i + ".YLabel.String = app.UnitsDropDown\_2.Value;");
344. end
345. end
346. % Dochádza k zmenám hodnoty: AddUnitsEditField
347. function AddUnitsEditFieldValueChanging(app, event)
348. uiprogressdlg(app.DescriptiveStatistic, 'Title', "Please Wait", "Indeterminate", "on");
349. % Pri písaní overuje počet znakov, pokiaľ je kolónka na prvej záložke prázdna,
350. % zablokuje tlačidlo Add.:
351. if size(event.Value) > 0
352. app.AddButton.Enable = true;
353. else
354. app.AddButton.Enable = false;
355. end
356. end
357. % Dochádza k zmenám hodnoty: AddUnitsEditField\_2
358. function AddUnitsEditField\_2ValueChanging(app, event)
359. uiprogressdlg(app.DescriptiveStatistic, 'Title', "Please Wait", "Indeterminate", "on");
360. % Pri písani overuje počet znakov, pokiaľ je kolónka na druhej záložke prázdna,
361. % zablokuje tlačidlo Add.:
362. if size(event.Value) > 0
363. app.AddButton\_2.Enable = true;
364. else
365. app.AddButton\_2.Enable = false;
366. end
367. end
368. % Funkcia stlačenia tlačidla: AddButton
369. function AddButtonPushed(app, event)
370. % Funkcia pridá do listu textový reťazec umiestnený v kolónke Add Units prvej záložky
371. % a následne ho odstráni z kolónky.:
372. uiprogressdlg(app.DescriptiveStatistic, 'Title', "Please Wait", "Indeterminate", "on");
373. app.UnitsDropDown.Items = [app.UnitsDropDown.Items, app.AddUnitsEditField.Value];
374. app.AddUnitsEditField.Value = "";
375. app.AddButton.Enable = false;
376. end
377. % Funkcia stlačenia tlačidla: AddButton\_2
378. function AddButton\_2Pushed(app, event)
379. % Funkcia pridá do listu textový reťazec umiestnený v kolónke Add Units druhej záložky
380. % a následne ho odstráni z kolónky.:
381. uiprogressdlg(app.DescriptiveStatistic, 'Title', "Please Wait", "Indeterminate", "on");
382. app.UnitsDropDown\_2.Items = [app.UnitsDropDown\_2.Items, app.AddUnitsEditField\_2.Value];
383. app.AddUnitsEditField\_2.Value = "";
384. app.AddButton\_2.Enable = false;
385. end

B.) Pomocné funkcie

1. % DESCRIPTIVE STATISTICS (ZÁLOŽKA\_1)
2. % Funkcia ds\_objects\_visibility.:
3. % Funkcia slúži na hromadné zakrývanie, blokovanie, odkrývanie a odblokovanie všetkých
4. % objektov prvej záložky aplikácie.:
5. function ds\_objects\_visibility(app, input\_boolean)
6. OBJECTS\_VE = [
7. app.YearDropDown;
8. app.YearDropDownLabel;
9. app.DateDropDown;
10. app.DateDropDownLabel;
11. app.PreviousButton;
12. app.NextButton;
13. app.UnitsDropDown;
14. app.UnitsDropDownLabel;
15. app.AddUnitsEditField;
16. app.AddUnitsLabel;
17. app.Switch;
18. app.WeekdayLabel;
19. ];
20. OBJECTS\_V = [
21. app.AddButton;
22. app.UIAxes\_1;
23. app.UIAxes\_2;
24. app.UIAxes\_3;
25. app.UIAxes\_4;
26. app.UIAxes\_5;
27. app.UIAxes\_6;
28. app.UIAxes\_7;
29. app.UIAxes\_8;
30. app.UIAxes\_9;
31. app.UIAxes\_10;
32. ];
33. set(OBJECTS\_VE, 'Visible', input\_boolean);
34. set(OBJECTS\_VE, 'Enable', input\_boolean);
35. set(OBJECTS\_V, 'Visible', input\_boolean);
36. end
37. % Funkcia ds\_gui\_visibility.:
38. % Funkcia slúži na hromadné zakrývanie, blokovanie, odkrývanie a odblokovanie objektov
39. % zobrazovacej plochy prvej záložky aplikácie. To znamená, všetkých objektov okrem objektov
40. % hlavného ovládacieho panela.:
41. function ds\_gui\_visibility(app, input\_boolean)
42. if ~isempty(app.UIAxes\_1.Children)
43. set(app.UIAxes\_1.Children, 'Visible', input\_boolean);
44. set(app.UIAxes\_4.Children, 'Visible', input\_boolean);
45. set(app.UIAxes\_8.Children, 'Visible', input\_boolean);
46. set(app.UIAxes\_10.Children, 'Visible', input\_boolean);
47. for i = 1:10
48. if i <= 3
49. set(app.UIAxes\_2.Children(i), 'Visible', input\_boolean);
50. set(app.UIAxes\_3.Children(i), 'Visible', input\_boolean);
51. set(app.UIAxes\_5.Children(i), 'Visible', input\_boolean);
52. end
53. if i <= 2
54. set(app.UIAxes\_6.Children(i), 'Visible', input\_boolean);
55. set(app.UIAxes\_7.Children(i), 'Visible', input\_boolean);
56. set(app.UIAxes\_9.Children(i), 'Visible', input\_boolean);
57. end
58. if i >= 2 && i <= 10
59. eval("legend(app.UIAxes\_" + i + ", 'hide');");
60. end
61. end
62. end
63. end
64. % Funkcia ds\_update.:
65. % Funkcia slúži na aktualizáciu aktuálneho zobrazenia prvej záložky aplikácie.:
66. function ds\_update(app, idx\_begin, idx\_end)
67. % Táto časť funkcie definuje často opakujúce sa premenné.:
68. AX = app.UIAxes\_1;
69. START = app.Stats\_1.DateStart(idx\_begin);
70. STOP = app.Stats\_1.DateStop(idx\_end) + seconds(1);
71. % Táto časť funkcie aktualizuje objekt Weekday.:
72. [~, app.WeekdayLabel.Text] = weekday(START, 'long');
73. % Táto časť funkcie aktualizuje hlavný signál údajov meraného dňa.:
74. x\_axis = app.Values\_1.Time(timerange(START, STOP));
75. x\_axis = string(x\_axis, 'HH:mm:ss');
76. x\_axis = duration(x\_axis, 'InputFormat', 'hh:mm:ss', 'Format', 'hh:mm');
77. y\_axis = app.Values\_1.Value(timerange(START, STOP));
78. plot(AX, x\_axis, y\_axis, 'k', 'LineWidth', 1);
79. ylabel(AX, app.UnitsDropDown.Value);
80. offset = AX.YLim(1);
81. range = AX.YLim(2) - offset;
82. ymin = offset - (0.05 \* range);
83. ymax = (offset + range) + (0.05 \* range);
84. ylim(AX, [ymin, ymax]);
85. axis(AX, 'tight');
86. % Táto časť funkcie aktualizuje ostatné grafy signálov štatistických parametrov.:
87. ds\_plot\_series(app, app.UIAxes\_2, [idx\_begin, idx\_end], ["Maximum", "Mean", "Minimum"], [1, 2, 3], 0.6, true);
88. ds\_plot\_series(app, app.UIAxes\_3, [idx\_begin, idx\_end], ["Mode", "Median", "Mean"], [1, 3, 2], 0.6, true);
89. ds\_plot\_series(app, app.UIAxes\_5, [idx\_begin, idx\_end], ["Variance", "SD", "IQR"], [1, 2, 3], 0.6, true);
90. ds\_plot\_series(app, app.UIAxes\_6, [idx\_begin, idx\_end], ["Range", "IQR"], [4, 3], 0.6, true);
91. ds\_plot\_series(app, app.UIAxes\_7, [idx\_begin, idx\_end], ["Variance", "RSD"], [1, 5], 0.6, true);
92. ds\_plot\_series(app, app.UIAxes\_8, [idx\_begin, idx\_end], "Skewness", 1, 0.6, false);
93. ds\_plot\_series(app, app.UIAxes\_9, [idx\_begin, idx\_end], ["Skewness", "Kurtosis"], [1, 2], 0.6, false);
94. ds\_plot\_series(app, app.UIAxes\_10, [idx\_begin, idx\_end], "Kurtosis", 2, 0.6, false);
95. % Táto časť funkcie aktualizuje bodový graf rovnosti parametrov centrálnej tendencie.:
96. x\_stats = string(app.Stats\_1{idx\_begin:idx\_end, "DateStart"}, 'HH:mm');
97. x\_stats = duration(x\_stats, 'InputFormat', 'hh:mm', 'Format', 'hh:mm');
98. y\_equality = app.Stats\_1.Equality(idx\_begin:idx\_end);
99. scatter(app.UIAxes\_4, x\_stats, y\_equality, 'filled');
100. ylabel(app.UIAxes\_4, app.UnitsDropDown.Value);
101. xlim(app.UIAxes\_4, [x\_stats(1), x\_stats(end)]);
102. ylim(app.UIAxes\_4, [0, 0.06]);
103. % Táto časť funkcie aktualizuje tlačidla na posun medzi dňami.:
104. is\_year\_first = app.YearDropDown.Value == string(app.YearDropDown.Items{1});
105. is\_year\_last = app.YearDropDown.Value == string(app.YearDropDown.Items{end});
106. is\_date\_first = app.DateDropDown.Value == string(app.DateDropDown.Items{end});
107. is\_date\_last = app.DateDropDown.Value == string(app.DateDropDown.Items{1});
108. if is\_year\_first && is\_date\_first
109. app.PreviousButton.Enable = false;
110. app.NextButton.Enable = true;
111. elseif is\_year\_last && is\_date\_last
112. app.PreviousButton.Enable = true;
113. app.NextButton.Enable = false;
114. else
115. app.PreviousButton.Enable = true;
116. app.NextButton.Enable = true;
117. end
118. % Táto časť funkcie aktualizuje prepínač legiend grafov.:
119. app.SwitchValueChanged(app);
120. end
121. % Funkcia ds\_load\_dates.:
122. % Funkcia slúži na odčítanie zvoleného dátumu z rozbaľovacích listov prvej záložky
123. % aplikácie.:
124. function dates = ds\_load\_dates(app)
125. dates\_days = string(unique(datetime( ...
126. app.Stats\_1.DateStart.Year, ...
127. app.Stats\_1.DateStart.Month, ...
128. app.Stats\_1.DateStart.Day, ...
129. "Format", "dd. MMMM")));
130. dates = ["Select"; flip(dates\_days)];
131. end
132. % Funkcia ds\_find\_index.:
133. % Funkcia slúži na odčítanie indexu zvoleného dátumu z rozbaľovacích listov prvej záložky
134. % aplikácie. Index je rovný prvému a poslednému riadku v tabuľke štatistických parametrov.
135. % Všetky riadky v tomto rozsahu vrátane prislúchajú zvolenému dňu.:
136. function idx = ds\_find\_index(app, select\_date\_drop\_down\_object)
137. selected\_date = get(select\_date\_drop\_down\_object, 'Value');
138. selected\_date = datetime(selected\_date, 'Format', 'dd. MMMM', 'InputFormat', 'dd. MMMM');
139. all\_dates = datetime(app.Stats\_1.DateStart, 'Format', 'dd. MMMM', "InputFormat", 'dd. MMMM');
140. match = ismember(string(all\_dates), string(selected\_date));
141. index\_begin = find(match, 1, 'first');
142. index\_end = find(match, 1, 'last');
143. idx = [index\_begin, index\_end];
144. end
145. % Funkcia ds\_plot\_series.:
146. % Funkcia slúži na vykreslenie jedného grafu prvej záložky aplikácie.:
147. function ds\_plot\_series(app, ax, idx, y\_data, color, opacity, ylabel\_bool)
148. x\_axis = string(app.Stats\_1{idx(1):idx(2), "DateStart"}, 'HH:mm');
149. x\_axis = duration(x\_axis, 'InputFormat', 'hh:mm', 'Format', 'hh:mm');
150. for i = 1:size(y\_data, 2)
151. y\_axis = app.Stats\_1{idx(1):idx(2), y\_data(i)};
152. plot(ax, x\_axis, y\_axis, 'Color', [app.Palette(color(i), :), opacity], 'LineWidth', 1);
153. if i == 1
154. hold(ax, "on");
155. end
156. end
157. if ylabel\_bool
158. ylabel(ax, app.UnitsDropDown.Value);
159. end
160. axis(ax, 'tight');
161. offset = ax.YLim(1);
162. range = ax.YLim(2) - offset;
163. ymin = offset - (0.05 \* range);
164. ymax = (offset + range) + (0.05 \* range);
165. ylim(ax, [ymin, ymax]);
166. hold(ax, "off");
167. end
168. % PLOTS OF STATISTICS (ZÁLOŽKA\_2)
169. % Funkcia pos\_objects\_visibility.:
170. % Funkcia slúži na hromadné zakrývanie, blokovanie, odkrývanie a odblokovanie všetkých
171. % objektov druhej záložky aplikácie.:
172. function pos\_objects\_visibility(app, input\_boolean)
173. OBJECTS = [
174. app.YearDropDown\_2;
175. app.YearDropDown\_2Label;
176. app.DateDropDown\_2;
177. app.DateDropDown\_2Label;
178. app.PreviousButton\_2;
179. app.NextButton\_2;
180. app.UnitsDropDown\_2;
181. app.UnitsDropDown\_2Label;
182. app.AddUnitsEditField\_2;
183. app.AddUnitsEditField\_2Label;
184. app.Switch\_2;
185. app.WeekdayLabel\_2;
186. ];
187. PLOTS = [
188. app.UIAxes\_11;
189. app.UIAxes\_12;
190. app.UIAxes\_13;
191. app.UIAxes\_14;
192. app.UIAxes\_15;
193. app.UIAxes\_16;
194. app.UIAxes\_17;
195. app.UIAxes\_18;
196. app.UIAxes\_19;
197. app.UIAxes\_20;
198. ];
199. set(app.AddButton\_2, 'Visible', input\_boolean);
200. set(OBJECTS, 'Visible', input\_boolean);
201. set(OBJECTS, 'Enable', input\_boolean);
202. set(PLOTS, 'Visible', input\_boolean);
203. end
204. % Funkcia pos\_gui\_visibility.:
205. % Funkcia slúži na hromadné zakrývanie, blokovanie, odkrývanie a odblokovanie objektov
206. % zobrazovacej plochy druhej záložky aplikácie. To znamená, všetkých objektov okrem
207. % objektov hlavného ovládacieho panela.:
208. function pos\_gui\_visibility(app, input\_boolean)
209. if ~isempty(app.UIAxes\_11.Children)
210. set(app.UIAxes\_11.Children, 'Visible', input\_boolean);
211. for i = 12:20
212. eval("set(app.UIAxes\_" + i + ".Children(1), 'Visible', input\_boolean);");
213. eval("set(app.UIAxes\_" + i + ".Children(2), 'Visible', input\_boolean);");
214. eval("legend(app.UIAxes\_" + i + ",'hide');");
215. end
216. end
217. end
218. % Funkcia pos\_update.:
219. % Funkcia slúži na aktualizáciu aktuálneho zobrazenia druhej záložky aplikácie.:
220. function pos\_update(app, idx\_begin, idx\_end)
221. % Táto časť funkcie definuje často opakujúce sa premenné.:
222. AX = app.UIAxes\_11;
223. START = app.Stats\_2.DateStart(idx\_begin);
224. STOP = app.Stats\_2.DateStop(idx\_end) + seconds(1);
225. % Táto časť funkcie aktualizuje objekt Weekday.:
226. [~, app.WeekdayLabel\_2.Text] = weekday(START, 'long');
227. % Táto časť funkcie aktualizuje hlavný signál údajov meraného dňa.:
228. x\_axis = app.Values\_2.Time(timerange(START, STOP));
229. x\_axis = string(x\_axis, 'HH:mm:ss');
230. x\_axis = duration(x\_axis, 'InputFormat', 'hh:mm:ss', 'Format', 'hh:mm');
231. y\_axis = app.Values\_2.Value(timerange(START, STOP));
232. plot(AX, x\_axis, y\_axis, 'k', 'LineWidth', 1);
233. ylabel(AX, app.UnitsDropDown\_2.Value);
234. offset = AX.YLim(1);
235. range = AX.YLim(2) - offset;
236. ymin = offset - (0.05 \* range);
237. ymax = (offset + range) + (0.05 \* range);
238. ylim(AX, [ymin, ymax]);
239. axis(AX, 'tight');
240. % Táto časť funkcie aktualizuje ostatné grafy signálov štatistických parametrov.:
241. idx = [idx\_begin, idx\_end];
242. pos\_plot\_series(app, app.UIAxes\_12, idx, ["Range", "SD"], [1, 2], 0.6, true);
243. pos\_plot\_series(app, app.UIAxes\_13, idx, ["Range", "Skewness"], [1, 3], 0.6, true);
244. pos\_plot\_series(app, app.UIAxes\_14, idx, ["Range", "Kurtosis"], [1, 4], 0.6, true);
245. pos\_plot\_series(app, app.UIAxes\_15, idx, ["Range", "AccuracyRate"], [1, 5], 0.6, true);
246. pos\_plot\_series(app, app.UIAxes\_16, idx, ["SD", "Skewness"], [2, 3], 0.6, true);
247. pos\_plot\_series(app, app.UIAxes\_17, idx, ["Kurtosis", "AccuracyRate"], [4, 5], 0.6, true);
248. pos\_plot\_series(app, app.UIAxes\_18, idx, ["SD", "AccuracyRate"], [2, 5], 0.6, true);
249. pos\_plot\_series(app, app.UIAxes\_19, idx, ["Skewness", "AccuracyRate"], [3, 5], 0.6, true);
250. pos\_plot\_series(app, app.UIAxes\_20, idx, ["SD", "Kurtosis"], [2, 4], 0.6, true);
251. % Táto časť funkcie aktualizuje tlačidla na posun medzi dňami.:
252. is\_year\_first = app.YearDropDown\_2.Value == string(app.YearDropDown\_2.Items{1});
253. is\_year\_last = app.YearDropDown\_2.Value == string(app.YearDropDown\_2.Items{end});
254. is\_date\_first = app.DateDropDown\_2.Value == string(app.DateDropDown\_2.Items{end});
255. is\_date\_last = app.DateDropDown\_2.Value == string(app.DateDropDown\_2.Items{1});
256. if is\_year\_first && is\_date\_first
257. app.PreviousButton\_2.Enable = false;
258. app.NextButton\_2.Enable = true;
259. elseif is\_year\_last && is\_date\_last
260. app.PreviousButton\_2.Enable = true;
261. app.NextButton\_2.Enable = false;
262. else
263. app.PreviousButton\_2.Enable = true;
264. app.NextButton\_2.Enable = true;
265. end
266. % Táto časť funkcie aktualizuje prepínač legiend grafov.:
267. Switch\_2ValueChanged(app);
268. end
269. % Funkcia pos\_load\_dates.:
270. % Funkcia slúži na odčítanie zvoleného dátumu z rozbaľovacích listov druhej záložky
271. % aplikácie.:
272. function dates = pos\_load\_dates(app)
273. dates\_days = string(unique(datetime( ...
274. app.Stats\_2.DateStart.Year, ...
275. app.Stats\_2.DateStart.Month, ...
276. app.Stats\_2.DateStart.Day, ...
277. "Format", "dd. MMMM")));
278. dates = ["Select"; flip(dates\_days)];
279. end
280. % Funkcia pos\_find\_index.:
281. % Funkcia slúži na odčítanie indexu zvoleného dátumu z rozbaľovacích listov druhej záložky
282. % aplikácie. Index je rovný prvému a poslednému riadku v tabuľke štatistických parametrov.
283. % Všetky riadky v tomto rozsahu vrátane prislúchajú zvolenému dňu.:
284. function idx = pos\_find\_index(app, select\_date\_drop\_down\_object)
285. selected\_date = get(select\_date\_drop\_down\_object, 'Value');
286. selected\_date = datetime(selected\_date, 'Format', 'dd. MMMM', 'InputFormat', 'dd. MMMM');
287. all\_dates = datetime(app.Stats\_2.DateStart, 'Format', 'dd. MMMM', "InputFormat", 'dd. MMMMM');
288. match = ismember(string(all\_dates), string(selected\_date));
289. index\_begin = find(match, 1, 'first');
290. index\_end = find(match, 1, 'last');
291. idx = [index\_begin, index\_end];
292. end
293. % Funkcia pos\_plot\_series.:
294. % Funkcia slúži na vykreslenie jedného grafu druhej záložky aplikácie.:
295. function pos\_plot\_series(app, ax, idx, y\_data, color, opacity, ylabel\_bool)
296. x\_axis = string(app.Stats\_2{idx(1):idx(2), "DateStart"}, 'HH:mm');
297. x\_axis = duration(x\_axis, 'InputFormat', 'hh:mm', 'Format', 'hh:mm');
298. for i = 1:size(y\_data, 2)
299. y\_axis = app.Stats\_2{idx(1):idx(2), y\_data(i)};
300. plot(ax, x\_axis, y\_axis, 'Color', [app.Palette(color(i), :), opacity], 'LineWidth', 1);
301. if i == 1
302. hold(ax, "on");
303. end
304. end
305. if ylabel\_bool
306. ylabel(ax, app.UnitsDropDown\_2.Value);
307. end
308. axis(ax, 'tight');
309. offset = ax.YLim(1);
310. range = ax.YLim(2) - offset;
311. ymin = offset - (0.05 \* range);
312. ymax = (offset + range) + (0.05 \* range);
313. ylim(ax, [ymin, ymax]);
314. hold(ax, "off");
315. end

C.) Vlastnosti aplikácie

1. % Vlastnosti aplikácie.:
2. properties (Access = private)
3. Values\_1 % Časová tabuľka čistých údajov pre záložku Descriptive Statistics.
4. Values\_2 % Časová tabuľka čistých údajov pre záložku Plots of Statistics.
5. Stats\_1 % Tabuľka štatistických parametrov pre záložku Descriptive Statistics.
6. Stats\_2 % Tabuľka štatistických parametrov pre záložku Plots of Statistics.
7. Palette % Paleta farieb pre vykreslené krivky.
8. Interval\_Length % Dĺžka intervalov v sekundách.
9. Signal\_Lengths % List dĺžok intervalov v minútach.
10. Years % List rokov.
11. CurrentYear % Zvolený rok.
12. End

**10. Kód aplikácie *Statistics\_Analysis***

A.) Funkcie spätných volaní

1. % Kód, ktorý sa vykoná hneď po otvorení aplikácie.:
2. function startupFcn(app)
3. % Nastaví aplikácii šírku (W) a výšku (H) v pixeloch a umiestni ju na stred obrazovky.:
4. W = 1920; H = 1080;
5. R = get(0, 'screensize');
6. app.Statistics\_AnalysisUIapp.Position = [(R(3) / 2) - (W / 2), (R(4) / 2) - (H / 2), W, H];
7. uiprogressdlg(app.Statistics\_AnalysisUIapp, 'Title', "Please Wait", "Indeterminate", "on");
8. % Zakryje všetky objekty v aplikácii a ponechá len prvý objekt.:
9. objects\_visibility(app, false);
10. % Načíta zo súboru years.mat list rokov a nahradí rozbaľovaci list týmito hodnotami.:
11. app.Years = load("years.mat", "-mat", 'YEARS\_STRINGS');
12. app.Years = app.Years.YEARS\_STRINGS;
13. app.YearDropDown.Items = ["Select"; app.Years];
14. % Načíta zo súboru signal\_lengths.mat list dĺžok signálov a nahradí rozbaľovacie listy
15. % týmito hodnotami.:
16. app.Signal\_Lengths = load("signal\_lengths.mat", "SIGNAL\_LENGTHS\_STRINGS");
17. app.Signal\_Lengths = app.Signal\_Lengths.SIGNAL\_LENGTHS\_STRINGS;
18. app.SignalLengthDropDown.Items = ["Select"; app.Signal\_Lengths];
19. end
20. % Došlo k zmene hodnoty: SignalLengthDropDown
21. function SignalLengthDropDownValueChanged(app, event)
22. uiprogressdlg(app.Statistics\_AnalysisUIapp, 'Title', "Please Wait", "Indeterminate", "on");
23. % Táto časť kódu zakryje všetky nasledujúce kroky výberu v prípade, že sú viditeľné.:
24. % Odstráni možnosť "Select" zo zoznamu pri prvom výbere.:
25. if app.SignalLengthDropDown.Items(1) == "Select"
26. app.SignalLengthDropDown.Items(1) = [];
27. else
28. app.YearDropDown.Items = ["Select"; app.Years];
29. app.YearDropDown.Value = app.YearDropDown.Items(1);
30. gui\_visibility(app, false);
31. end
32. % Zobrazí objekty SignalLengthDropDown a YearDropDown.:
33. objects\_visibility(app, false);
34. app.SignalLengthDropDown.Visible = true;
35. app.SignalLengthDropDown.Enable = true;
36. app.SignalLengthLabel.Visible = true;
37. app.SignalLengthLabel.Enable = true;
38. app.YearDropDown.Enable = true;
39. app.YearDropDown.Visible = true;
40. app.YearDropDownLabel.Enable = true;
41. app.YearDropDownLabel.Visible = true;
42. end
43. % Došlo k zmene hodnoty: YearDropDown
44. function YearDropDownValueChanged(app, event)
45. uiprogressdlg(app.Statistics\_AnalysisUIapp, 'Title', "Please Wait", "Indeterminate", "on");
46. app.CurrentYear = app.YearDropDown.Value;
47. % Načíta časovú tabuľku čistých údajov a uloží ju do vlastnosti aplikácie Values.:
48. app.Values = load("values\_clean\_" + app.CurrentYear + ".mat", "values\_clean\_" + app.CurrentYear);
49. eval("app.Values = app.Values.values\_clean\_" + app.CurrentYear + ";");
50. % Táto časť kódu zakryje všetky nasledujúce kroky výberu v prípade, že sú viditeľné.:
51. if app.YearDropDown.Items(1) == "Select"
52. app.YearDropDown.Items(1) = [];
53. else
54. app.DateDropDown.Items = ["Select", app.DateDropDown.Items];
55. app.DateDropDown.Value = app.DateDropDown.Items(1);
56. gui\_visibility(app, false);
57. end
58. % Načíta tabuľku vypočítaných štatistických parametrov a uloží ju do vlastnosti Stats.:
59. MINUTES = erase(app.SignalLengthDropDown.Value, " min");
60. MINUTES = str2double(MINUTES);
61. app.Interval\_Length = seconds(minutes(MINUTES));
62. app.Stats = load("stats\_" + app.CurrentYear + ".mat", "stats\_" + MINUTES + "\_" + app.CurrentYear);
63. eval("app.Stats = app.Stats.stats\_" + MINUTES + "\_" + app.CurrentYear + ";");
64. % Zobrazí objekty SignalLengthDropDown, YearDropDown a DateDropDown.:
65. objects\_visibility(app, false);
66. app.SignalLengthDropDown.Visible = true;
67. app.SignalLengthDropDown.Enable = true;
68. app.SignalLengthLabel.Visible = true;
69. app.SignalLengthLabel.Enable = true;
70. app.YearDropDown.Enable = true;
71. app.YearDropDown.Visible = true;
72. app.YearDropDownLabel.Enable = true;
73. app.YearDropDownLabel.Visible = true;
74. app.DateDropDown.Enable = true;
75. app.DateDropDown.Visible = true;
76. app.DateDropDownLabel.Enable = true;
77. app.DateDropDownLabel.Visible = true;
78. % Načíta list dostupných dátumov dní do rozbaľovacieho listu.:
79. app.DateDropDown.Items = load\_dates(app);
80. % Nastaví prvú hodnotu z listu dátumov, t.j. hodnota "Select".:
81. app.DateDropDown.Value = app.DateDropDown.Items(1);
82. end
83. % Došlo k zmene hodnoty: DateDropDown
84. function DateDropDownValueChanged(app, event)
85. uiprogressdlg(app.Statistics\_AnalysisUIapp, 'Title', "Please Wait", "Indeterminate", "on");
86. % Táto časť kódu zakryje všetky nasledujúce kroky výberu v prípade, že sú viditeľné.:
87. if app.DateDropDown.Items(1) == "Select"
88. app.DateDropDown.Items(1) = [];
89. else
90. app.TimeDropDown.Items = ["Select", app.TimeDropDown.Items];
91. app.TimeDropDown.Value = app.TimeDropDown.Items(1);
92. gui\_visibility(app, false);
93. end
94. % Zobrazí objekty SignalLengthDropDown, YearDropDown, DateDropDown a TimeDropDown.:
95. objects\_visibility(app, false);
96. app.SignalLengthDropDown.Visible = true;
97. app.SignalLengthDropDown.Enable = true;
98. app.SignalLengthLabel.Visible = true;
99. app.SignalLengthLabel.Enable = true;
100. app.YearDropDown.Enable = true;
101. app.YearDropDown.Visible = true;
102. app.YearDropDownLabel.Enable = true;
103. app.YearDropDownLabel.Visible = true;
104. app.DateDropDown.Enable = true;
105. app.DateDropDown.Visible = true;
106. app.DateDropDownLabel.Enable = true;
107. app.DateDropDownLabel.Visible = true;
108. app.TimeDropDown.Enable = true;
109. app.TimeDropDown.Visible = true;
110. app.TimeDropDownLabel.Enable = true;
111. app.TimeDropDownLabel.Visible = true;
112. % Načíta list dostupných časov dní do rozbaľovacieho listu.:
113. app.TimeDropDown.Items = load\_starts(app);
114. % Nastaví prvú hodnotu z listu časov, t.j. hodnota "Select".:
115. app.TimeDropDown.Value = app.TimeDropDown.Items(1);
116. end
117. % Došlo k zmene hodnoty: TimeDropDown
118. function TimeDropDownValueChanged(app, event)
119. uiprogressdlg(app.Statistics\_AnalysisUIapp, 'Title', "Please Wait", "Indeterminate", "on");
120. % Ak bol výber z listu časov vykonaný prvýkrát, odstráni z listu prvú možnosť
121. % "Select".:
122. if app.TimeDropDown.Items(1) == "Select"
123. app.TimeDropDown.Items(1) = [];
124. end
125. % Zobrazí všetky ostatné objekty aplikácie:
126. objects\_visibility(app, true);
127. % Zavolá sa najprv pomocná funkcia find\_index na vyhľadanie indexu v závislosti od
128. % zvoleného dňa. Následne sa zavolá hlavná funkcia update na aktualizáciu
129. % zobrazenia.:
130. Index = find\_index(app, app.DateDropDown, app.TimeDropDown);
131. update(app, Index);
132. end
133. % Funkcia stlačenia tlačidla: PreviousButton
134. function PreviousButtonPushed(app, event)
135. uiprogressdlg(app.Statistics\_AnalysisUIapp, 'Title', "Please Wait", "Indeterminate", "on");
136. % Funkcia uloží pozície aktuálneho výberu všetkých rozbaľovacích listov.:
137. idx\_year = find(ismember(app.YearDropDown.Items, app.YearDropDown.Value), 1);
138. idx\_date = find(ismember(app.DateDropDown.Items, app.DateDropDown.Value), 1);
139. idx\_time = find(ismember(app.TimeDropDown.Items, app.TimeDropDown.Value), 1);
140. % Pomocné logické premenné, ktoré overia, či je aktuálne zvolená prvá hodnota v
141. % listoch.:
142. is\_date\_first = idx\_date == size(app.DateDropDown.Items, 2);
143. is\_time\_first = idx\_time == 1;
144. % Ak je zvolený prvý deň v liste dátumov a zároveň prvý čas v liste časov, zobrazenie
145. % sa presunie automaticky do predchádzajúceho roku poslednej hodiny posledného
146. % dostupného dňa.:
147. if is\_time\_first && is\_date\_first
148. app.YearDropDown.Value = app.YearDropDown.Items(idx\_year - 1);
149. YearDropDownValueChanged(app);
150. app.DateDropDown.Value = app.DateDropDown.Items(2);
151. DateDropDownValueChanged(app);
152. app.TimeDropDown.Value = app.TimeDropDown.Items(end);
153. TimeDropDownValueChanged(app);
154. % Ak je zvolený prvý čas v liste časov a zároveň nie je zvolený prvý deň v liste
155. % dátumov, zobrazenie sa presunie automaticky do poslednej hodiny predchádzajúceho
156. % dňa.:
157. elseif is\_time\_first && ~is\_date\_first
158. app.DateDropDown.Value = app.DateDropDown.Items(idx\_date + 1);
159. DateDropDownValueChanged(app);
160. app.TimeDropDown.Value = app.TimeDropDown.Items(end);
161. TimeDropDownValueChanged(app);
162. % Ak nie je zvolený prvý čas z listu časov, posunie sa zobrazenie na predchádzajúci čas
163. % rovnakého dňa.:
164. elseif ~is\_time\_first
165. app.TimeDropDown.Value = app.TimeDropDown.Items(idx\_time - 1);
166. TimeDropDownValueChanged(app);
167. end
168. end
169. % Funkcia stlačenia tlačidla: NextButton
170. function NextButtonPushed(app, event)
171. uiprogressdlg(app.Statistics\_AnalysisUIapp, 'Title', "Please Wait", "Indeterminate", "on");
172. % Funkcia uloží pozície aktuálneho výberu všetkých rozbaľovacích listov.:
173. idx\_year = find(ismember(app.YearDropDown.Items, app.YearDropDown.Value), 1);
174. idx\_date = find(ismember(app.DateDropDown.Items, app.DateDropDown.Value), 1);
175. idx\_time = find(ismember(app.TimeDropDown.Items, app.TimeDropDown.Value), 1);
176. % Pomocné logické premenné, ktoré overia, či je aktuálne zvolená posledná hodnota v
177. % listoch.:
178. is\_date\_last = idx\_date == 1;
179. is\_time\_last = idx\_time == size(app.TimeDropDown.Items, 2);
180. % Ak je zvolený posledný deň v liste dátumov a zároveň posledný čas v liste časov,
181. % zobrazenie sa presunie automaticky do následujúceho roku prvej hodiny prvého
182. % dostupného dňa.:
183. if is\_time\_last && is\_date\_last
184. app.YearDropDown.Value = app.YearDropDown.Items(idx\_year + 1);
185. YearDropDownValueChanged(app);
186. app.DateDropDown.Value = app.DateDropDown.Items(end);
187. DateDropDownValueChanged(app);
188. app.TimeDropDown.Value = app.TimeDropDown.Items(2);
189. TimeDropDownValueChanged(app);
190. % Ak je zvolený posledný čas v liste časov a zároveň nie je zvolený posledný deň
191. % v liste dátumov, zobrazenie sa presunie automaticky do prvej hodiny následujúceho
192. % dňa.:
193. elseif is\_time\_last && ~is\_date\_last
194. app.DateDropDown.Value = app.DateDropDown.Items(idx\_date - 1);
195. DateDropDownValueChanged(app);
196. app.TimeDropDown.Value = app.TimeDropDown.Items(2);
197. TimeDropDownValueChanged(app);
198. % Ak nie je zvolený posledný čas z listu časov, posunie sa zobrazenie na následujúci
199. % čas rovnakého dňa.:
200. elseif ~is\_time\_last
201. app.TimeDropDown.Value = app.TimeDropDown.Items(idx\_time + 1);
202. TimeDropDownValueChanged(app);
203. end
204. end
205. % Došlo k zmene hodnoty: UnitsDropDown
206. function UnitsDropDownValueChanged(app, event)
207. uiprogressdlg(app.Statistics\_AnalysisUIapp, 'Title', "Please Wait", "Indeterminate", "on");
208. value = app.UnitsDropDown.Value;
209. % Pokiaľ používateľ zmení jednotky z listu jednotiek, zmenia sa jednotky na všetkých
210. % príslušných grafoch zobrazenia.:
211. app.Axes\_Signal.YLabel.String = value;
212. app.Axes\_PDF.XLabel.String = value;
213. app.Axes\_CDF.XLabel.String = value;
214. end
215. % Dochádza k zmenám hodnôt: AddUnitsEditField
216. function AddUnitsEditFieldValueChanging(app, event)
217. uiprogressdlg(app.Statistics\_AnalysisUIapp, 'Title', "Please Wait", "Indeterminate", "on");
218. % Pri písaní overuje počet znakov, pokiaľ je kolónka prázdna, zablokuje tlačidlo Add.:
219. if size(event.Value) > 0
220. app.AddButton.Enable = 1;
221. else
222. app.AddButton.Enable = 0;
223. end
224. end
225. % Funkcia stlačenia tlačidla: AddButton
226. function AddButtonPushed(app, event)
227. uiprogressdlg(app.Statistics\_AnalysisUIapp, 'Title', "Please Wait", "Indeterminate", "on");
228. % Funkcia pridá do listu textový reťazec umiestnený v kolónke Add Units a následne ho
229. % odstráni z kolónky.:
230. app.UnitsDropDown.Items = [app.UnitsDropDown.Items, app.AddUnitsEditField.Value];
231. app.AddUnitsEditField.Value = "";
232. app.AddButton.Enable = 0;
233. End

B.) Pomocné funkcie

1. % Funkcia objects\_visibility.:
2. % Funkcia slúži na hromadné zakrývanie, blokovanie, odkrývanie a odblokovanie všetkých
3. % objektov aplikácie.:
4. function objects\_visibility(app, input\_boolean)
5. OBJECTS\_VE = [
6. app.YearDropDown;
7. app.YearDropDownLabel;
8. app.DateDropDown;
9. app.DateDropDownLabel;
10. app.TimeDropDown;
11. app.TimeDropDownLabel;
12. app.PreviousButton;
13. app.NextButton;
14. app.UnitsDropDown;
15. app.UnitsDropDownLabel;
16. app.AddUnitsEditField;
17. app.AddUnitsLabel;
18. app.WeekdayLabel;
19. ];
20. OBJECTS\_V = [
21. app.AddButton;
22. app.Axes\_Signal;
23. app.Axes\_CDF;
24. app.Axes\_PDF;
25. app.Axes\_PP\_Plot;
26. ];
27. set(OBJECTS\_VE, 'Visible', input\_boolean);
28. set(OBJECTS\_VE, 'Enable', input\_boolean);
29. set(OBJECTS\_V, 'Visible', input\_boolean);
30. if input\_boolean == true
31. set(app.StatsTable, 'Visible', 'on');
32. set(app.StatsTable, 'Enable', 'on');
33. else
34. set(app.StatsTable, 'Visible', 'off');
35. set(app.StatsTable, 'Enable', 'off');
36. end
37. end
38. % Funkcia gui\_visibility.:
39. function gui\_visibility(app, input\_boolean)
40. if ~isempty(app.Axes\_Signal.Children)
41. set(app.Axes\_Signal.Children, 'Visible', input\_boolean);
42. set(app.Axes\_CDF.Children, 'Visible', input\_boolean);
43. set(app.Axes\_PDF.Children, 'Visible', input\_boolean);
44. set(app.Axes\_PP\_Plot.Children, 'Visible', input\_boolean);
45. end
46. if input\_boolean == true
47. set(app.StatsTable, 'Visible', 'on');
48. set(app.StatsTable, 'Enable', 'on');
49. else
50. set(app.StatsTable, 'Visible', 'off');
51. set(app.StatsTable, 'Enable', 'off');
52. end
53. end
54. % Funkcia load\_dates.:
55. % Funkcia slúži na načítanie listu dátumov z tabuľky štatistických parametrov.:
56. function dates = load\_dates(app)
57. dates\_days = string(unique(datetime( ...
58. app.Stats.DateStart.Year, ...
59. app.Stats.DateStart.Month, ...
60. app.Stats.DateStart.Day, ...
61. "Format", "dd. MMMM")));
62. dates = ["Select"; flip(dates\_days)];
63. end
64. % Funkcia load\_starts.:
65. % Na základe zvoleného dátumu funkcia vygeneruje list dostupných časov merania, v ktorých
66. % začína meraná vzorka.:
67. function starts = load\_starts(app)
68. selected\_date = app.DateDropDown.Value;
69. selected\_date = datetime(selected\_date, 'Format', 'dd. MMMM', 'InputFormat', 'dd. MMMM');
70. all\_dates = datetime(app.Stats.DateStart, 'Format', 'dd. MMMM', "InputFormat", 'dd. MMMM');
71. match = ismember(string(all\_dates), string(selected\_date));
72. starts = string(app.Stats.DateStart(match), 'HH:mm');
73. starts = ["Select"; starts];
74. end
75. % Funkcia find\_index.:
76. % Funkcia slúži na odčítanie indexu zvoleného dátumu z rozbaľovacích listov aplikácie.
77. % Prvý index je rovný prvému a druhý poslednému riadku v tabuľke štatistických parametrov.
78. % Všetky riadky v tomto rozsahu vrátane prislúchajú zvolenému dňu.:
79. function idx = find\_index(app, date\_dropdown, time\_dropdown)
80. selected\_date = get(date\_dropdown, 'Value');
81. selected\_date = datetime(selected\_date, 'Format', "dd. MMMM", 'InputFormat', "dd. MMMM");
82. selected\_time = get(time\_dropdown, 'Value');
83. selected\_time = datetime(selected\_time, 'Format', "HH:mm", 'InputFormat', "HH:mm");
84. selected\_datetime = selected\_date + hours(selected\_time.Hour) + minutes(selected\_time.Minute);
85. selected\_datetime = datetime(selected\_datetime, 'Format', "dd. MMMM HH:mm");
86. app.CurrentDatetime = selected\_datetime;
87. all\_dates = datetime(app.Stats.DateStart, 'Format', "dd. MMMM HH:mm", "InputFormat", "dd. MMMM HH:mm");
88. match = ismember(string(all\_dates), string(selected\_datetime));
89. idx = find(match, 1);
90. end
91. % Funkcia update.:
92. function update(app, idx)
93. % Táto časť funkcie aktualizuje objekt Weekday.:
94. [~, app.WeekdayLabel.Text] = weekday(app.Stats.DateStart(idx), 'long');
95. % Táto časť funkcie definuje často opakujúce sa premenné.:
96. START = app.Stats.DateStart(idx);
97. STOP = app.Stats.DateStop(idx) + seconds(1);
98. INTERVAL = timerange(START, STOP);
99. MIN = app.Stats.Minimum(idx);
100. MAX = app.Stats.Maximum(idx);
101. % Nastavuje krok horizontálnej osi signálového grafu podľa zvolenej dĺžky signálu.:
102. app.StatsTable.Data = rows2vars(app.Stats(idx, :));
103. % Táto časť kódu zaisťuje správne vykreslenie grafu funkcie hustoty pravdepodobnosti
104. % (PDF).:
105. x1 = MIN:0.1:MAX;
106. mu = app.Stats.Mean(idx);
107. sigma = app.Stats.SD(idx);
108. y1 = pdf("Normal", x1, mu, sigma);
109. plot(app.Axes\_PDF, x1, y1);
110. hold(app.Axes\_PDF, 'on');
111. scatter(app.Axes\_PDF, x1, y1, 'blue');
112. hold(app.Axes\_PDF, 'off');
113. axis(app.Axes\_PDF, "tight");
114. % Táto časť kódu zaisťuje, že graf kumulatívnej distribúcie (CDF) je správne
115. % vykreslený.:
116. x2 = MIN:0.1:MAX;
117. mu = app.Stats.Mean(idx);
118. sigma = app.Stats.SD(idx);
119. y2 = cdf("Normal", x2, mu, sigma);
120. plot(app.Axes\_CDF, x2, y2);
121. hold(app.Axes\_CDF, 'on');
122. scatter(app.Axes\_CDF, x2, y2, 'blue');
123. hold(app.Axes\_CDF, 'off');
124. axis(app.Axes\_CDF, "tight");
125. % Táto časť kódu zabezpečuje správne vykreslenie grafu časového priebehu.:
126. X = app.Values.Time(INTERVAL);
127. Y = app.Values.Value(INTERVAL);
128. plot(app.Axes\_Signal, X, Y);
129. axis(app.Axes\_Signal, 'tight');
130. ylim(app.Axes\_Signal, [app.Axes\_Signal.YLim(1) - 0.1, app.Axes\_Signal.YLim(2) + 0.1]);
131. app.Axes\_Signal.XLim = [START, STOP];
132. % Táto časť kódu zabezpečuje, že graf overenia intervalov spoľahlivosti je vykreslený
133. % správne.:
134. a = table();
135. a.probs = (0:0.001:0.999)';
136. a.z\_input = (a.probs + 1) / 2;
137. a.z = norminv(a.z\_input);
138. a.left = mu - (a.z \* sigma);
139. a.right = mu + (a.z \* sigma);
140. a.inside = sum(Y >= a.left' & Y <= a.right')';
141. a.inside\_perc = (a.inside / app.Interval\_Length) \* 100;
142. x1\_res = a.probs \* 100;
143. y1\_res = a.probs \* 100;
144. x2\_res = a.probs \* 100;
145. y2\_res = a.inside\_perc;
146. plot(app.Axes\_PP\_Plot, x1\_res, y1\_res, "black");
147. hold(app.Axes\_PP\_Plot, "on");
148. grid(app.Axes\_PP\_Plot, "on");
149. scatter(app.Axes\_PP\_Plot, x2\_res, y2\_res, 10, "red", "filled", "MarkerFaceAlpha", 0.33, "MarkerEdgeAlpha", 0.33);
150. axis(app.Axes\_PP\_Plot, "square");
151. hold(app.Axes\_PP\_Plot, "off");
152. UnitsDropDownValueChanged(app);
153. % Zakáže tlačidlá Ďalej alebo Späť, keď je v aplikácii momentálne zobrazená prvá
154. % alebo posledná vzorka.:
155. is\_year\_first = app.YearDropDown.Value == string(app.YearDropDown.Items{1});
156. is\_year\_last = app.YearDropDown.Value == string(app.YearDropDown.Items{end});
157. is\_date\_first = app.DateDropDown.Value == string(app.DateDropDown.Items{end});
158. is\_date\_last = app.DateDropDown.Value == string(app.DateDropDown.Items{1});
159. is\_time\_first = app.TimeDropDown.Value == string(app.TimeDropDown.Items{1});
160. is\_time\_last = app.TimeDropDown.Value == string(app.TimeDropDown.Items{end});
161. if is\_year\_first && is\_date\_first && is\_time\_first
162. app.PreviousButton.Enable = false;
163. app.NextButton.Enable = true;
164. elseif is\_year\_last && is\_date\_last && is\_time\_last
165. app.PreviousButton.Enable = true;
166. app.NextButton.Enable = false;
167. else
168. app.PreviousButton.Enable = true;
169. app.NextButton.Enable = true;
170. end
171. end

C.) Vlastnosti aplikácie

1. % Vlastnosti aplikácie.:
2. properties (Access = private)
3. Values % Časová tabuľka čistých údajov.
4. Stats % Tabuľka štatistických parametrov.
5. Interval\_Length % Dĺžka intervalov v sekundách.
6. Signal\_Lengths % List dĺžok intervalov v minútach.
7. Years % List rokov.
8. CurrentYear % Zvolený rok.
9. CurrentDatetime % Zvolený dátum a čas.
10. End

**11. Kód aplikácie *Signals***

A.) Funkcie spätných volaní

1. % Kód, ktorý sa vykoná hneď po otvorení aplikácie.:
2. function startupFcn(app)
3. % Nastaví aplikácii šírku (W) a výšku (H) v pixeloch a umiestni ju na stred obrazovky.:
4. W = 1920; H = 1080;
5. R = get(0, 'screensize');
6. app.SignalsUIFigure.Position = [(R(3) / 2) - (W / 2), (R(4) / 2) - (H / 2), W, H];
7. uiprogressdlg(app.SignalsUIFigure, 'Title', "Please Wait", "Indeterminate", "on");
8. % Zakryje všetky objekty v aplikácii a ponechá len prvý objekt.:
9. objects\_visibility(app, false);
10. % Načíta zo súboru years.mat list rokov a nahradí rozbaľovaci list týmito hodnotami.:
11. app.Years = load("years.mat", "-mat", 'YEARS\_STRINGS');
12. app.Years = app.Years.YEARS\_STRINGS;
13. app.YearDropDown.Items = ["Select"; app.Years];
14. % Načíta zo súboru signal\_lengths.mat list dĺžok signálov a nahradí rozbaľovacie listy
15. % týmito hodnotami.:
16. app.Signal\_Lengths = load("signal\_lengths.mat", "SIGNAL\_LENGTHS\_STRINGS");
17. app.Signal\_Lengths = app.Signal\_Lengths.SIGNAL\_LENGTHS\_STRINGS;
18. app.SignalLengthDropDown.Items = ["Select"; app.Signal\_Lengths];
19. % Pridelí vlastnosti tejto aplikácie s názvom Graphs objekty grafických okien.:
20. app.Graphs = [app.Axes\_1, app.Axes\_2, app.Axes\_3, app.Axes\_4];
21. end
22. % Došlo k zmene hodnoty: SignalLengthDropDown
23. function SignalLengthDropDownValueChanged(app, event)
24. uiprogressdlg(app.SignalsUIFigure, 'Title', "Please Wait", "Indeterminate", "on");
25. % Táto časť kódu zakryje všetky nasledujúce kroky výberu v prípade, že sú viditeľné.:
26. % Odstráni možnosť "Select" zo zoznamu pri prvom výbere.:
27. if app.SignalLengthDropDown.Items(1) == "Select"
28. app.SignalLengthDropDown.Items(1) = [];
29. else
30. if app.YearDropDown.Items(1) ~= "Select"
31. app.YearDropDown.Items = ["Select", app.YearDropDown.Items];
32. app.YearDropDown.Value = app.YearDropDown.Items(1);
33. end
34. if ~isempty(app.Axes\_1.Children)
35. app.Axes\_1.Children.Visible = "off";
36. app.Axes\_2.Children.Visible = "off";
37. app.Axes\_3.Children.Visible = "off";
38. app.Axes\_4.Children.Visible = "off";
39. end
40. end
41. % Zobrazí objekty SignalLengthDropDown a YearDropDown.:
42. objects\_visibility(app, false);
43. app.SignalLengthDropDown.Visible = true;
44. app.SignalLengthDropDown.Enable = true;
45. app.SignalLengthDropDownLabel.Visible = true;
46. app.SignalLengthDropDownLabel.Enable = true;
47. app.YearDropDown.Enable = true;
48. app.YearDropDown.Visible = true;
49. app.YearDropDownLabel.Enable = true;
50. app.YearDropDownLabel.Visible = true;
51. end
52. % Došlo k zmene hodnoty: YearDropDown
53. function YearDropDownValueChanged(app, event)
54. uiprogressdlg(app.SignalsUIFigure, 'Title', "Please Wait", "Indeterminate", "on");
55. % Načíta časovú tabuľku čistých údajov a uloží ju do vlastnosti aplikácie Values.:
56. app.CurrentYear = app.YearDropDown.Value;
57. app.Values = load("values\_clean\_" + app.CurrentYear + ".mat", "values\_clean\_" + app.CurrentYear);
58. eval("app.Values = app.Values.values\_clean\_" + app.CurrentYear + ";");
59. % Načíta tabuľku vypočítaných štatistických parametrov a uloží ju do vlastnosti Stats.:
60. MINUTES = erase(app.SignalLengthDropDown.Value, " min");
61. MINUTES = str2double(MINUTES);
62. app.Interval\_Length = seconds(minutes(MINUTES));
63. app.Stats = load("stats\_" + app.CurrentYear + ".mat", "stats\_" + MINUTES + "\_" + app.CurrentYear);
64. eval("app.Stats = app.Stats.stats\_" + MINUTES + "\_" + app.CurrentYear + ";");
65. % Táto časť kódu zakryje všetky nasledujúce kroky výberu v prípade, že sú viditeľné.:
66. if app.YearDropDown.Items(1) == "Select"
67. app.YearDropDown.Items(1) = [];
68. else
69. if ~isempty(app.Axes\_1.Children)
70. app.Axes\_1.Children.Visible = "off";
71. app.Axes\_2.Children.Visible = "off";
72. app.Axes\_3.Children.Visible = "off";
73. app.Axes\_4.Children.Visible = "off";
74. end
75. end
76. % Zobrazí objekty SignalLengthDropDown, YearDropDown a DateDropDown.:
77. objects\_visibility(app, false);
78. app.SignalLengthDropDown.Visible = true;
79. app.SignalLengthDropDown.Enable = true;
80. app.SignalLengthDropDownLabel.Visible = true;
81. app.SignalLengthDropDownLabel.Enable = true;
82. app.YearDropDown.Enable = true;
83. app.YearDropDown.Visible = true;
84. app.YearDropDownLabel.Enable = true;
85. app.YearDropDownLabel.Visible = true;
86. app.DateDropDown.Enable = true;
87. app.DateDropDown.Visible = true;
88. app.DateDropDownLabel.Enable = true;
89. app.DateDropDownLabel.Visible = true;
90. % Načíta list dostupných dátumov dní do rozbaľovacieho listu.:
91. app.DateDropDown.Items = load\_dates(app);
92. % Nastaví prvú hodnotu z listu dátumov, t.j. hodnota "Select".:
93. app.DateDropDown.Value = app.DateDropDown.Items(1);
94. end
95. % Došlo k zmene hodnoty: DateDropDown
96. function DateDropDownValueChanged(app, event)
97. uiprogressdlg(app.SignalsUIFigure, 'Title', "Please Wait", "Indeterminate", "on");
98. % Táto časť kódu zakryje všetky nasledujúce kroky výberu v prípade, že sú viditeľné.:
99. if app.DateDropDown.Items(1) == "Select"
100. app.DateDropDown.Items(1) = [];
101. else
102. if ~isempty(app.Axes\_1.Children)
103. app.Axes\_1.Children.Visible = "off";
104. app.Axes\_2.Children.Visible = "off";
105. app.Axes\_3.Children.Visible = "off";
106. app.Axes\_4.Children.Visible = "off";
107. end
108. end
109. % Zobrazí objekty SignalLengthDropDown, YearDropDown, DateDropDown a TimeDropDown.:
110. objects\_visibility(app, false);
111. app.SignalLengthDropDown.Visible = true;
112. app.SignalLengthDropDown.Enable = true;
113. app.SignalLengthDropDownLabel.Visible = true;
114. app.SignalLengthDropDownLabel.Enable = true;
115. app.YearDropDown.Enable = true;
116. app.YearDropDown.Visible = true;
117. app.YearDropDownLabel.Enable = true;
118. app.YearDropDownLabel.Visible = true;
119. app.DateDropDown.Enable = true;
120. app.DateDropDown.Visible = true;
121. app.DateDropDownLabel.Enable = true;
122. app.DateDropDownLabel.Visible = true;
123. app.TimeDropDown.Enable = true;
124. app.TimeDropDown.Visible = true;
125. app.TimeDropDownLabel.Enable = true;
126. app.TimeDropDownLabel.Visible = true;
127. % Načíta list dostupných časov dní do rozbaľovacieho listu.:
128. app.TimeDropDown.Items = load\_starts(app);
129. % Nastaví prvú hodnotu z listu časov, t.j. hodnota "Select".:
130. app.TimeDropDown.Value = app.TimeDropDown.Items(1);
131. % Ak je v liste časov len jedná hodnota, t.j. hodnota "Select", vymaže tento čas
132. % z listu časov a zobrazí sa nasledujúci čas.:
133. if size(app.TimeDropDown.Items, 2) == 1
134. app.DateDropDown.Items(1) = [];
135. app.DateDropDown.Value = app.DateDropDown.Items(1);
136. DateDropDownValueChanged(app);
137. end
138. end
139. % Došlo k zmene hodnoty: TimeDropDown
140. function TimeDropDownValueChanged(app, event)
141. uiprogressdlg(app.SignalsUIFigure, 'Title', "Please Wait", "Indeterminate", "on");
142. % Ak bol výber z listu časov vykonaný prvýkrát, odstráni z listu prvú možnosť
143. % "Select".:
144. if app.TimeDropDown.Items(1) == "Select"
145. app.TimeDropDown.Items(1) = [];
146. end
147. % Zobrazí všetky ostatné objekty aplikácie:
148. objects\_visibility(app, true);
149. % Zavolá sa najprv pomocná funkcia find\_index na vyhľadanie indexu v závislosti od
150. % zvoleného dňa. Následne sa zavolá hlavná funkcia update na aktualizáciu
151. % zobrazenia.:
152. Index = find\_index(app, app.DateDropDown, app.TimeDropDown);
153. update(app, Index);
154. end
155. % Funkcia stlačenia tlačidla: PreviousButton
156. function PreviousButtonPushed(app, event)
157. uiprogressdlg(app.SignalsUIFigure, 'Title', "Please Wait", "Indeterminate", "on");
158. % Funkcia uloží pozície aktuálneho výberu všetkých rozbaľovacích listov.:
159. idx\_year = find(ismember(app.YearDropDown.Items, app.YearDropDown.Value), 1);
160. idx\_date = find(ismember(app.DateDropDown.Items, app.DateDropDown.Value), 1);
161. idx\_time = find(ismember(app.TimeDropDown.Items, app.TimeDropDown.Value), 1);
162. % Pomocné logické premenné, ktoré overia, či je aktuálne zvolená prvá hodnota v
163. % listoch.:
164. is\_date\_first = idx\_date == size(app.DateDropDown.Items, 2);
165. is\_time\_first = idx\_time == 1;
166. % Ak je zvolený prvý deň v liste dátumov a zároveň prvý čas v liste časov, zobrazenie
167. % sa presunie automaticky do predchádzajúceho roku poslednej hodiny posledného
168. % dostupného dňa.:
169. if is\_time\_first && is\_date\_first
170. app.YearDropDown.Value = app.YearDropDown.Items(idx\_year - 1);
171. YearDropDownValueChanged(app);
172. app.DateDropDown.Value = app.DateDropDown.Items(2);
173. DateDropDownValueChanged(app);
174. app.TimeDropDown.Value = app.TimeDropDown.Items(end);
175. TimeDropDownValueChanged(app);
176. % Ak je zvolený prvý čas v liste časov a zároveň nie je zvolený prvý deň v liste
177. % dátumov, zobrazenie sa presunie automaticky do poslednej hodiny predchádzajúceho
178. % dňa.:
179. elseif is\_time\_first && ~is\_date\_first
180. app.DateDropDown.Value = app.DateDropDown.Items(idx\_date + 1);
181. DateDropDownValueChanged(app);
182. app.TimeDropDown.Value = app.TimeDropDown.Items(end);
183. TimeDropDownValueChanged(app);
184. % Ak nie je zvolený prvý čas z listu časov, posunie sa zobrazenie na predchádzajúci čas
185. % rovnakého dňa.:
186. elseif ~is\_time\_first
187. app.TimeDropDown.Value = app.TimeDropDown.Items(idx\_time - 1);
188. TimeDropDownValueChanged(app);
189. end
190. end
191. % Funkcia stlačenia tlačidla: NextButton
192. function NextButtonPushed(app, event)
193. uiprogressdlg(app.SignalsUIFigure, 'Title', "Please Wait", "Indeterminate", "on");
194. % Funkcia uloží pozície aktuálneho výberu všetkých rozbaľovacích listov.:
195. idx\_year = find(ismember(app.YearDropDown.Items, app.YearDropDown.Value), 1);
196. idx\_date = find(ismember(app.DateDropDown.Items, app.DateDropDown.Value), 1);
197. idx\_time = find(ismember(app.TimeDropDown.Items, app.TimeDropDown.Value), 1);
198. % Pomocné logické premenné, ktoré overia, či je aktuálne zvolená posledná hodnota v
199. % listoch.:
200. is\_date\_last = idx\_date == 1;
201. is\_time\_last = idx\_time == size(app.TimeDropDown.Items, 2);
202. % Ak je zvolený posledný deň v liste dátumov a zároveň posledný čas v liste časov,
203. % zobrazenie sa presunie automaticky do následujúceho roku prvej hodiny prvého
204. % dostupného dňa.:
205. if is\_time\_last && is\_date\_last
206. app.YearDropDown.Value = app.YearDropDown.Items(idx\_year + 1);
207. YearDropDownValueChanged(app);
208. app.DateDropDown.Value = app.DateDropDown.Items(end);
209. DateDropDownValueChanged(app);
210. app.TimeDropDown.Value = app.TimeDropDown.Items(2);
211. TimeDropDownValueChanged(app);
212. % Ak je zvolený posledný čas v liste časov a zároveň nie je zvolený posledný deň
213. % v liste dátumov, zobrazenie sa presunie automaticky do prvej hodiny následujúceho
214. % dňa.:
215. elseif is\_time\_last && ~is\_date\_last
216. app.DateDropDown.Value = app.DateDropDown.Items(idx\_date - 1);
217. DateDropDownValueChanged(app);
218. app.TimeDropDown.Value = app.TimeDropDown.Items(2);
219. TimeDropDownValueChanged(app);
220. % Ak nie je zvolený posledný čas z listu časov, posunie sa zobrazenie na následujúci
221. % čas rovnakého dňa.:
222. elseif ~is\_time\_last
223. app.TimeDropDown.Value = app.TimeDropDown.Items(idx\_time + 1);
224. TimeDropDownValueChanged(app);
225. end
226. end
227. % Došlo k zmene hodnoty: UnitsDropDown
228. function UnitsDropDownValueChanged(app, event)
229. uiprogressdlg(app.SignalsUIFigure, 'Title', "Please Wait", "Indeterminate", "on");
230. % Pokiaľ používateľ zmení jednotky z listu jednotiek, zmenia sa jednotky na všetkých
231. % príslušných grafoch zobrazenia.:
232. for i = 1:4
233. app.Graphs(i).YLabel.String = app.UnitsDropDown.Value;
234. end
235. end
236. % Dochádza k zmenám hodnôt: AddUnitsEditField
237. function AddUnitsEditFieldValueChanging(app, event)
238. uiprogressdlg(app.SignalsUIFigure, 'Title', "Please Wait", "Indeterminate", "on");
239. % Pri písaní overuje počet znakov, pokiaľ je kolónka prázdna, zablokuje tlačidlo Add.:
240. if size(event.Value) > 0
241. app.AddButton.Enable = 1;
242. else
243. app.AddButton.Enable = 0;
244. end
245. end
246. % Funkcia stlačenia tlačidla: AddButton
247. function AddButtonPushed(app, event)
248. uiprogressdlg(app.SignalsUIFigure, 'Title', "Please Wait", "Indeterminate", "on");
249. % Funkcia pridá do listu textový reťazec umiestnený v kolónke Add Units a následne ho
250. % odstráni z kolónky.:
251. app.UnitsDropDown.Items = [app.UnitsDropDown.Items app.AddUnitsEditField.Value];
252. app.AddUnitsEditField.Value = "";
253. app.AddButton.Enable = 0;
254. end
255. % Výber sa zmenil: ButtonGroup
256. function ButtonGroupSelectionChanged(app, event)
257. uiprogressdlg(app.SignalsUIFigure, 'Title', "Please Wait", "Indeterminate", "on");
258. selectedButton = app.ButtonGroup.SelectedObject;
259. switch selectedButton.Text
260. % Hranice osi y grafov sú stanovené podľa minimálnej a maximálnej hodnoty
261. % vzťahujúcej sa na deň merania.:
262. case "1"
263. idx = find\_index(app, app.DateDropDown, app.TimeDropDown);
264. for i = 1:4
265. Index = app.Stats.DateStart.Day == app.CurrentDatetime.Day & ...
266. app.Stats.DateStart.Month == app.CurrentDatetime.Month;
267. if nnz(find(Index) == idx - 1 + i)
268. YLim\_Minimum = min(app.Stats.Minimum(Index));
269. YLim\_Maximum = max(app.Stats.Maximum(Index));
270. app.Graphs(i).YLim = [YLim\_Minimum - 0.1, YLim\_Maximum + 0.1];
271. else
272. Index = app.Stats.DateStart.Day == app.CurrentDatetime.Day + 1 & ...
273. app.Stats.DateStart.Month == app.CurrentDatetime.Month;
274. YLim\_Minimum = min(app.Stats.Minimum(Index));
275. YLim\_Maximum = max(app.Stats.Maximum(Index));
276. app.Graphs(i).YLim = [YLim\_Minimum - 0.1, YLim\_Maximum + 0.1];
277. end
278. end
279. % Hranice grafov na osi y sú stanovené podľa minimálnych a maximálnych hodnôt
280. % týkajúcich sa konkrétneho grafu.:
281. case "2"
282. for i = 1:4
283. axis(app.Graphs(i), "tight");
284. ylim(app.Graphs(i), [app.Graphs(i).YLim(1) - 0.1, app.Graphs(i).YLim(2) + 0.1]);
285. end
286. % Hranice grafov na osi y sú stanovené podľa minimálnej a maximálnej hodnoty
287. % aktuálneho výberu.:
288. case "3"
289. % Reset y-axis limits.
290. YLims = cell(4, 1);
291. for i = 1:4
292. ylim(app.Graphs(i), "auto");
293. YLims{i} = app.Graphs(i).YLim;
294. end
295. YLims = cell2mat(YLims);
296. YLim\_Minimum = min(YLims(:, 1));
297. YLim\_Maximum = max(YLims(:, 2));
298. for i = 1:4
299. app.Graphs(i).YLim = [YLim\_Minimum - 0.1, YLim\_Maximum + 0.1];
300. end
301. end
302. end

B.) Pomocné funkcie

1. % Funkcia objects\_visibility.:
2. % Funkcia slúži na hromadné zakrývanie, blokovanie, odkrývanie a odblokovanie všetkých
3. % objektov aplikácie.:
4. function objects\_visibility(app, input\_boolean)
5. OBJECTS\_VE = [
6. app.YearDropDown;
7. app.YearDropDownLabel;
8. app.DateDropDown;
9. app.DateDropDownLabel;
10. app.TimeDropDown;
11. app.TimeDropDownLabel;
12. app.PreviousButton;
13. app.NextButton;
14. app.Button\_1;
15. app.Button\_2;
16. app.Button\_3;
17. app.UnitsDropDown;
18. app.UnitsDropDownLabel;
19. app.AddUnitsEditField;
20. app.AddUnitsLabel;
21. app.WeekdayLabel;
22. ];
23. OBJECTS\_V = [
24. app.AddButton;
25. app.ButtonGroup;
26. app.Axes\_1;
27. app.Axes\_2;
28. app.Axes\_3;
29. app.Axes\_4;
30. ];
31. set(OBJECTS\_VE, 'Visible', input\_boolean);
32. set(OBJECTS\_VE, 'Enable', input\_boolean);
33. set(OBJECTS\_V, 'Visible', input\_boolean);
34. end
35. % Funkcia load\_dates.:
36. % Funkcia slúži na načítanie listu dátumov z tabuľky štatistických parametrov.:
37. function dates = load\_dates(app)
38. dates\_days = string(unique(datetime(...
39. app.Stats.DateStart.Year,...
40. app.Stats.DateStart.Month,...
41. app.Stats.DateStart.Day,...
42. 'Format', "dd. MMMM")));
43. dates = ["Select"; flip(dates\_days)];
44. end
45. % Funkcia load\_starts.:
46. % Na základe zvoleného dátumu funkcia vygeneruje list dostupných časov merania, v ktorých
47. % začína meraná vzorka.:
48. function starts = load\_starts(app)
49. selected\_date = app.DateDropDown.Value;
50. selected\_date = datetime(selected\_date, 'Format', 'dd. MMMM', 'InputFormat', 'dd. MMMM');
51. all\_dates = datetime(app.Stats.DateStart, 'Format', 'dd. MMMM', "InputFormat", 'dd. MMMM');
52. match = ismember(string(all\_dates), string(selected\_date));
53. starts = string(app.Stats.DateStart(match), 'HH:mm');
54. % Ak je zvolený posledný deň posledného dostupného roku merania, funkcia odstráni
55. % z listu posledné 3 záznamy, pretože sa v jeden moment zobrazujú okrem vybraného
56. % záznamu ďalšie 3 nasledujúce záznamy.:
57. if selected\_date == app.DateDropDown.Items{1}
58. if size(starts, 1) > 3
59. starts = starts(1:(end - 3));
60. else
61. starts = [];
62. end
63. end
64. starts = ["Select"; starts];
65. end
66. % Funkcia find\_index.:
67. % Funkcia slúži na odčítanie indexu zvoleného dátumu z rozbaľovacích listov aplikácie.
68. % Prvý index je rovný prvému a druhý poslednému riadku v tabuľke štatistických parametrov.
69. % Všetky riadky v tomto rozsahu vrátane prislúchajú zvolenému dňu.:
70. function idx = find\_index(app, date\_dropdown, time\_dropdown)
71. selected\_date = get(date\_dropdown, 'Value');
72. selected\_date = datetime(selected\_date, 'Format', "dd. MMMM", 'InputFormat', "dd. MMMM");
73. selected\_time = get(time\_dropdown, 'Value');
74. selected\_time = datetime(selected\_time, 'Format', "HH:mm", 'InputFormat', "HH:mm");
75. selected\_datetime = selected\_date + hours(selected\_time.Hour) + minutes(selected\_time.Minute);
76. selected\_datetime = datetime(selected\_datetime, 'Format', "dd. MMMM HH:mm");
77. app.CurrentDatetime = selected\_datetime;
78. all\_dates = datetime(app.Stats.DateStart, 'Format', "dd. MMMM HH:mm", "InputFormat", "dd. MMMM HH:mm");
79. match = ismember(string(all\_dates), string(selected\_datetime));
80. idx = find(match, 1);
81. end
82. % Funkcia update.:
83. % Funkcia slúži na aktualizáciu aktuálneho zobrazenia aplikácie.:
84. function update(app, idx)
85. % Táto časť funkcie aktualizuje objekt Weekday.:
86. [~, app.WeekdayLabel.Text] = weekday(app.Stats.DateStart(idx), 'long');
87. % Pre-alokácia buniek pre osi štyroch grafov.:
88. X = cell(1, 4);
89. Y = cell(1, 4);
90. % Cyklus, ktorý extrahuje údaje z databázy a vykresli časové priebehy pre štvoricu
91. % grafov.:
92. for i = 0:3
93. START = app.Stats.DateStart(idx + i);
94. STOP = app.Stats.DateStop(idx + i) + seconds(1);
95. X{i + 1} = app.Values.Time(timerange(START, STOP));
96. Y{i + 1} = app.Values.Value(timerange(START, STOP));
97. plot(app.Graphs(i + 1), X{i + 1}, Y{i + 1});
98. axis(app.Graphs(i + 1), 'tight');
99. End
100. % Aktualizácia režimu zobrazenia.:
101. ButtonGroupSelectionChanged(app);
102. % Aktualizácia jednotiek.:
103. UnitsDropDownValueChanged(app);
104. % Kontrola tlačidiel. Ak je zvolený prvý, resp. posledný záznam z databázy údajov,
105. % zablokuje sa príslušné tlačidlo na posun medzi záznamami.:
106. is\_year\_first = app.YearDropDown.Value == string(app.YearDropDown.Items{1});
107. is\_year\_last = app.YearDropDown.Value == string(app.YearDropDown.Items{end});
108. is\_date\_first = app.DateDropDown.Value == string(app.DateDropDown.Items{end});
109. is\_date\_last = app.DateDropDown.Value == string(app.DateDropDown.Items{1});
110. is\_time\_first = app.TimeDropDown.Value == string(app.TimeDropDown.Items{1});
111. is\_time\_last = app.TimeDropDown.Value == string(app.TimeDropDown.Items{end});
112. if is\_year\_first && is\_date\_first && is\_time\_first
113. app.PreviousButton.Enable = false;
114. app.NextButton.Enable = true;
115. elseif is\_year\_last && is\_date\_last && is\_time\_last
116. app.PreviousButton.Enable = true;
117. app.NextButton.Enable = false;
118. else
119. app.PreviousButton.Enable = true;
120. app.NextButton.Enable = true;
121. end
122. end

C.) Vlastnosti aplikácie

1. % Vlastnosti aplikácie.:
2. properties (Access = private)
3. Values % Časová tabuľka čistých údajov.
4. Stats % Tabuľka štatistických parametrov.
5. Graphs % Objekt, v ktorom sú umiestnené 4 grafické prvky Axes.
6. Interval\_Length % Dĺžka intervalov v sekundách.
7. Signal\_Lengths % List dĺžok intervalov v minútach.
8. Years % List rokov.
9. CurrentYear % Zvolený rok.
10. CurrentDatetime % Zvolený dátum a čas.
11. end

**12. Kód funkcie *plot\_correlation\_diagram***

1. function plot\_correlation\_diagram(varargin)
2. % Vstupné argumenty:
3. % STATS = Tabuľka štatistických parametrov.
4. % DATE = Pole začiatočného dátumu. Prvý numerický údaj predstavuje
5. % deň, druhý údaj predstavuje mesiac.
6. % VARIABLES = 2 Premenné tabuľky štatistických parametrov.
7. % N = Počet dní, ktoré budú vykreslené. Maximálne 8 ilustrácii.
8. % EQUAL\_AXIS = Logická hodnota, ktorou rozhodujeme, či osi všetkých grafov
9. % majú totožné, resp. globálne hranice.
10. STATS = varargin{1};
11. VARIABLES = varargin{2};
12. DATE = varargin{3};
13. N = varargin{4};
14. EQUAL\_AXIS = varargin{5};
15. % Táto časť funkcie extrahuje z názvu vstupnej tabuľky štatistických
16. % parametrov rok. Následne extrahuje z poľa DATE deň a mesiac. Následne
17. % vytvorí z týchto 3 údajov dátum, od ktorého sa spustí vykresľovanie N
18. % počtu grafov.:
19. STATS\_NAME = inputname(1);
20. YEAR = str2double(STATS\_NAME(10:end));
21. MONTH = DATE(2);
22. DAY = DATE(1);
23. DATE = datetime(YEAR, MONTH, DAY, 'Format', 'dd. MM. uuuu');
24. % Ak je omylom zadaný väčší počet grafov ako 8, prepíše túto hodnotu na
25. % maximálny počet grafov = 8.:
26. if N > 8
27. N = 8;
28. end
29. % Hlavný cyklus, ktorý postupne vykresľuje grafy.:
30. i = 0;
31. plot\_data\_all = [];
32. while i ~= N
33. % Extrahuje údaje z tabuľky štatistických hodnôt na základe dňa a
34. % mesiaca.:
35. MONTH = DATE.Month;
36. DAY = DATE.Day;
37. cond\_1 = STATS.DateStart.Day == DAY;
38. cond\_2 = STATS.DateStart.Month == MONTH;
39. plot\_data = STATS(cond\_1 & cond\_2, VARIABLES);
40. plot\_data\_all = [plot\_data\_all; plot\_data]; %#ok<AGROW>
41. % Ak cyklus prejde na posledný dostupný deň, ukončí vykresľovanie.:
42. if DATE.Year ~= YEAR
43. break;
44. end
45. % Ak neexistujú údaje pre tento deň, cyklus prejde do nasledujúceho
46. % dňa.:
47. if isempty(plot\_data)
48. DATE = DATE + days(1);
49. continue;
50. end
51. % Vygeneruje nové (prázdne) okno grafu.:
52. figure('Name', datestr(DATE), ...
53. 'WindowStyle', 'docked');
54. % Vykresli graf.:
55. s = scatter(plot\_data{:, 1}, ...
56. plot\_data{:, 2}, ...
57. 'filled', ...
58. 'LineWidth', 0.2, ...
59. 'MarkerFaceAlpha', 0.4);
60. % Nastavenie mriežky, názvu grafov a osi a režim zobrazenia "štvorec".:
61. grid('on');
62. title({"Correlation diagram between " + VARIABLES{1} + " and " ...
63. + VARIABLES{2}, ...
64. "Date: " + datestr(DATE, 'dd.mm. yyyy')});
65. xlabel(VARIABLES{1});
66. ylabel(VARIABLES{2});
67. s.DataTipTemplate.DataTipRows(1).Label = VARIABLES{1};
68. s.DataTipTemplate.DataTipRows(2).Label = VARIABLES{2};
69. axis('square');
70. axis('tight');
71. box('on');
72. % Pripraví nasledujúci cyklus do nasledujúceho dňa.:
73. i = i + 1;
74. DATE = DATE + days(1);
75. end
76. % Ak je vstupný argument EQUAL\_AXIS pravdivý (t.j. 1 alebo true),
77. % všetky grafy budú mať totožné hranice oboch osí.:
78. if EQUAL\_AXIS == true
79. x\_min = min(plot\_data\_all{:, 1});
80. x\_max = max(plot\_data\_all{:, 1});
81. y\_min = min(plot\_data\_all{:, 2});
82. y\_max = max(plot\_data\_all{:, 2});
83. for fig = 1:i
84. xlim(figure(fig).Children, [x\_min, x\_max]);
85. ylim(figure(fig).Children, [y\_min, y\_max]);
86. end
87. end
88. end

**13. Kód funkcie *plot\_correlation\_weekdays***

1. function plot\_correlation\_weekdays(varargin)
2. % Vstupné argumenty:
3. % STATS = Tabuľka štatistických parametrov.
4. % VARIABLE = Premenná (stĺpec) tabuľky STATS. Napríklad 'Mean' pre
5. % aritmetický priemer.
6. % SELECTION = Pole výberu. Prvá numerická hodnota predstavuje deň v
7. % týždni zvoleného mesiaca. Napríklad 5 pre piatok. Druhá
8. % numerická hodnota predstavuje mesiac v roku. Napríklad 2
9. % pre február.
10. STATS = varargin{1};
11. VARIABLE = varargin{2};
12. SELECTION = varargin{3};
13. % Táto časť kódu pripraví pole dátumov dní days na základe vstupu.:
14. WEEK\_DAY = SELECTION(1);
15. MONTH = SELECTION(2);
16. weekdays = weekday(STATS.DateStart) - 1;
17. weekdays(weekdays == 0) = 7;
18. cond\_1 = weekdays == WEEK\_DAY;
19. cond\_2 = STATS.DateStart.Month == MONTH;
20. tab = STATS(cond\_1 & cond\_2, {'DateStart', VARIABLE});
21. days = unique(tab.DateStart.Day);
22. days\_count = length(days);
23. final\_tab = cell(days\_count, 1);
24. % V tejto časti kódu sa vyplní tabuľka final\_tab všetkými údajmi potrebnými
25. % k vypočítaniu všetkých korelácií.:
26. idx = 1;
27. for i = 1:days\_count
28. final\_tab{idx, 1} = tab(tab.DateStart.Day == days(i), :);
29. idx = idx + 1;
30. end
31. % V tejto časti kódu sa vypočítajú všetky korelácie medzi štatistickým
32. % parametrom jednotlivých dni daného mesiaca, čím vygeneruje maticu
33. % výsledkov s názvom final\_corr.:
34. final\_tab\_length = length(final\_tab);
35. final\_corr = zeros(final\_tab\_length);
36. labels = NaT(final\_tab\_length, 1);
37. for col = 1:final\_tab\_length
38. labels(col) = datetime(final\_tab{col}.DateStart(1));
39. for row = 1:final\_tab\_length
40. data\_1 = final\_tab{row}{:, 2};
41. data\_2 = final\_tab{col}{:, 2};
42. if length(data\_1) == length(data\_2)
43. final\_corr(row, col) = corr(data\_1, data\_2);
44. else
45. final\_corr(row, col) = NaN;
46. end
47. end
48. end
49. % V tejto časti kódu funkcia vykresli maticu výsledných korelácií.:
50. labels = datetime(labels, 'Format', 'd. M. uuuu');
51. [~, week\_day\_label] = weekday(labels(1), 'long');
52. month\_label = datetime(labels(1), 'Format', 'MMMM');
53. hm = heatmap(final\_corr);
54. hm.XDisplayLabels = labels;
55. hm.YDisplayLabels = labels;
56. hm.Title = {
57. "Variable: " + VARIABLE;
58. "Week day: " + week\_day\_label;
59. "Month: " + string(month\_label);
60. };
61. end

**14. Kód funkcie *plot\_signal\_2D\_maps***

1. function plot\_signal\_2D\_maps(varargin)
2. % Vstupné argumenty:
3. % CLEAN = Časová tabuľka čistých údajov.
4. % DATE = Pole začiatočného dátumu a času. Prvý numerický údaj
5. % predstavuje deň, druhý údaj predstavuje mesiac, tretí
6. % hodinu a štvrtý minútu.
7. % N = Počet po sebe idúcich vzoriek, ktoré budú vykreslené.
8. % Maximálne 8 ilustrácii.
9. % EQUAL\_AXIS = Logická hodnota, ktorou rozhodujeme, či osi všetkých grafov
10. % majú totožné, resp. globálne hranice.
11. CLEAN = varargin{1};
12. INTERVAL = varargin{2};
13. DATE = varargin{3};
14. N = varargin{4};
15. EQUAL\_AXIS = varargin{5};
16. % Táto časť funkcie extrahuje z názvu vstupnej časovej tabuľky čistých
17. % údajov rok. Následne extrahuje z poľa DATE deň, mesiac, hodinu a minútu.
18. % Následne vytvorí z týchto 5 údajov dátum, od ktorého sa spustí
19. % vykresľovanie N počtu grafov.:
20. CLEAN\_NAME = inputname(1);
21. YEAR = str2double(CLEAN\_NAME(14:end));
22. DAY = DATE(1);
23. MONTH = DATE(2);
24. HOUR = DATE(3);
25. MINUTE = DATE(4);
26. DATE = datetime(YEAR, MONTH, DAY, HOUR, MINUTE, 0, ...
27. 'Format', 'dd.MM.uuuu HH:mm');
28. % Ak je omylom zadaný väčší počet grafov ako 8, prepíše túto hodnotu na
29. % maximálny počet grafov = 8.:
30. if N > 8
31. N = 8;
32. end
33. % Hlavný cyklus, ktorý postupne vykresľuje grafy.:
34. i = 0;
35. plot\_data\_all = [];
36. while i ~= N
37. % Extrahuje údaje z časovej tabu¾ky čistých hodnôt na základe dňa,
38. % mesiaca, hodiny a minúty.:
39. YEAR\_NEW = DATE.Year;
40. MONTH = DATE.Month;
41. DAY = DATE.Day;
42. HOUR = DATE.Hour;
43. MINUTE = DATE.Minute;
44. START = datetime(YEAR\_NEW, MONTH, DAY, HOUR, MINUTE, 0, ...
45. 'Format', 'dd.MM.uuuu HH:mm');
46. STOP = START + minutes(INTERVAL);
47. plot\_data = [CLEAN.Value(timerange(START, STOP)), ...
48. CLEAN.Value(timerange(START + seconds(1), STOP + seconds(1)))];
49. plot\_data\_all = [plot\_data\_all; plot\_data]; %#ok<AGROW>
50. % Ak cyklus prejde na poslednú dostupnú vzorku v roku, ukončí
51. % vykresľovanie.:
52. if DATE.Year ~= YEAR\_NEW
53. break;
54. end
55. % Ak neexistujú údaje pre túto vzorku, cyklus prejde do nasledujúcej
56. % vzorky.:
57. if isempty(plot\_data)
58. DATE = DATE + minutes(INTERVAL);
59. continue;
60. end
61. % Vygeneruje nové (prázdne) okno grafu.:
62. figure('Name', datestr(DATE), ...
63. 'WindowStyle', 'docked');
64. % Vykresli graf.:
65. s = scatter(plot\_data(:, 1), ...
66. plot\_data(:, 2), ...
67. 'filled', ...
68. 'LineWidth', 0.2, ...
69. 'MarkerFaceAlpha', 0.3);
70. % Nastavenie mriežky, názvu grafov a osi a režim zobrazenia "štvorec".:
71. grid('on');
72. title({"2D Map", ...
73. "Start: " + datestr(DATE, 'dd.mm. yyyy hh:MM'), ...
74. "Length: " + INTERVAL + " minutes"});
75. xlabel("Value");
76. ylabel("Value");
77. s.DataTipTemplate.DataTipRows(1).Label = "Value (N)";
78. s.DataTipTemplate.DataTipRows(2).Label = "Value (N+1)";
79. axis('square');
80. axis('tight');
81. box('on');
82. % Pripraví nasledujúci cyklus do nasledujúcej vzorky.:
83. i = i + 1;
84. DATE = DATE + minutes(INTERVAL);
85. end
86. % Ak je vstupný argument EQUAL\_AXIS pravdivý (t.j. 1 alebo true),
87. % všetky grafy budú mať totožné hranice oboch osí.:
88. if EQUAL\_AXIS == true
89. x\_min = min(plot\_data\_all(:, 1));
90. x\_max = max(plot\_data\_all(:, 1));
91. y\_min = min(plot\_data\_all(:, 2));
92. y\_max = max(plot\_data\_all(:, 2));
93. for fig = 1:i
94. xlim(figure(fig).Children, [x\_min, x\_max]);
95. ylim(figure(fig).Children, [y\_min, y\_max]);
96. end
97. end
98. end

**15. Kód funkcie *plot\_signal\_FFT***

1. function plot\_signal\_FFT(varargin)
2. % Vstupné argumenty:
3. % CLEAN = Časová tabuľka čistých hodnôt.
4. % INTERVAL = Dĺžka jednej vzorky údajov v minútach.
5. % FREQ = Frekvencia nameraného signálu v Hertzoch.
6. % DATE = Pole začiatočného dátumu a času. Prvý numerický údaj
7. % predstavuje deň, druhý údaj predstavuje mesiac, tretí
8. % hodinu a štvrtý minútu.
9. % N = Počet po sebe idúcich vzoriek, ktoré budú vykreslené.
10. % Maximálne 8 ilustrácii.
11. CLEAN = varargin{1};
12. INTERVAL = varargin{2};
13. FREQ = varargin{3};
14. DATE = varargin{4};
15. N = varargin{5};
16. % Táto časť funkcie extrahuje z názvu vstupnej časovej tabuľky čistých
17. % údajov rok. Následne extrahuje z poľa DATE deň, mesiac, hodinu a minútu.
18. % Následne vytvorí z týchto 5 údajov dátum, od ktorého sa spustí
19. % vykresľovanie N počtu grafov.:
20. CLEAN\_NAME = inputname(1);
21. YEAR = str2double(CLEAN\_NAME(14:end));
22. DAY = DATE(1);
23. MONTH = DATE(2);
24. HOUR = DATE(3);
25. MINUTE = DATE(4);
26. DATE = datetime(YEAR, MONTH, DAY, HOUR, MINUTE, 0, ...
27. 'Format', 'dd.MM.uuuu HH:mm');
28. % Ak je omylom zadaný väčší počet grafov ako 8, prepíše túto hodnotu na
29. % maximálny počet grafov = 8.:
30. if N > 8
31. N = 8;
32. end
33. % Hlavný cyklus, ktorý postupne vykresľuje grafy.:
34. i = 0;
35. while i ~= N
36. % Extrahuje údaje z časovej tabuľky čistých hodnôt na základe dňa,
37. % mesiaca, hodiny a minúty.:
38. YEAR\_NEW = DATE.Year;
39. MONTH = DATE.Month;
40. DAY = DATE.Day;
41. HOUR = DATE.Hour;
42. MINUTE = DATE.Minute;
43. START = datetime(YEAR\_NEW, MONTH, DAY, HOUR, MINUTE, 0, ...
44. 'Format', 'dd.MM.uuuu HH:mm');
45. STOP = START + minutes(INTERVAL);
46. plot\_data = table( ...
47. CLEAN.Time(timerange(START, STOP)), ...
48. CLEAN.Value(timerange(START, STOP)));
49. % Ak cyklus prejde na poslednú dostupnú vzorku v roku, ukončí
50. % vykresľovanie.:
51. if DATE.Year ~= YEAR\_NEW
52. break;
53. end
54. % Ak neexistujú údaje pre túto vzorku, cyklus prejde do nasledujúcej
55. % vzorky.:
56. if isempty(plot\_data)
57. DATE = DATE + minutes(INTERVAL);
58. continue;
59. end
60. % Vygeneruje nové (prázdne) okno grafu.:
61. figure('Name', datestr(DATE), ...
62. 'WindowStyle', 'docked');
63. % Vykresli graf signálu.:
64. subplot(2, 1, 1);
65. p1 = plot(plot\_data.Var1, plot\_data.Var2);
66. % Nastavenie mriežky, názvu grafov a osi a režim zobrazenia "štvorec".:
67. grid('on');
68. title({"Signal", ...
69. "Start: " + datestr(DATE, 'dd.mm. yyyy hh:MM'), ...
70. "Length: " + INTERVAL + " minutes"});
71. ylabel("Value");
72. p1.DataTipTemplate.DataTipRows(1).Label = "Time";
73. p1.DataTipTemplate.DataTipRows(2).Label = "Value";
74. axis('tight');
75. box('on');
76. % Vykresli graf FFT.:
77. subplot(2, 1, 2);
78. LENGTH = length(plot\_data.Var2);
79. COUNT = 2 ^ nextpow2(LENGTH);
80. half = COUNT / 2;
81. FFT = fft(plot\_data.Var2, COUNT);
82. x\_fft = (1:COUNT) \* (FREQ / COUNT);
83. y\_fft = abs(FFT) / COUNT;
84. p2 = plot(x\_fft(2:half), y\_fft(2:half));
85. % Nastavenie mriežky, názvu grafov a osi a režim zobrazenia "štvorec".:
86. grid('on');
87. title("Single-Sided Fast Fourier Transformation");
88. xlabel("Frequency (Hz)");
89. ylabel("|X(\omegaj)|");
90. p2.DataTipTemplate.DataTipRows(1).Label = "Frequency (Hz)";
91. p2.DataTipTemplate.DataTipRows(2).Label = "|X(\omegaj)|";
92. axis('tight');
93. box('on');
94. % Pripraví nasledujúci cyklus do nasledujúceho dňa.:
95. i = i + 1;
96. DATE = DATE + minutes(INTERVAL);
97. end
98. end

**17. Kód funkcie *folder\_processing***

1. function folder\_processing(varargin)
2. %% Funkcia folder\_processing.:
3. % Vstupné argumenty funkcie.:
4. % FOLDER = Cesta k priečinku, v ktorom sa nachádza aspoň jeden taký
5. % pod-priečinok, kde sú uložené údajové súbory s \*.edf príponou.
6. close all;
7. FOLDER = varargin{1};
8. subfolders = dir(FOLDER);
9. subfolders = struct2table(subfolders);
10. subfolders = string(subfolders.name(subfolders.isdir));
11. subfolders = subfolders(subfolders ~= "." & subfolders ~= "..");
12. % Cyklus, ktorý spracuje postupne každý pod-priečinok. Najprv vytvorí
13. % dočasnú tabuľku údajov, následne uloží grafy časových priebehov
14. % teplomerov a priebehy prvej diferenciálnej funkcie vektora časových
15. % značiek. Vygeneruje a uloží údaje vo formáte \*.csv do prislušných
16. % pod-priečinkov.:
17. for i = 1:length(subfolders)
18. path = FOLDER + "\" + subfolders{i};
19. cd(path);
20. disp(i + "/" + length(subfolders))
21. [H, T] = create\_table\_thermometer(path);
22. fig = plot\_all\_thermometers(T, "Value");
23. if ~isempty(fig)
24. saveas(fig, 'Value\_T.fig');
25. saveas(fig, 'Value\_T.png');
26. end
27. fig = plot\_all\_thermometers(H, "Value");
28. if ~isempty(fig)
29. saveas(fig, 'Value\_RH.fig');
30. saveas(fig, 'Value\_RH.png');
31. end
32. fig = plot\_all\_thermometers(T, "Diff\_1");
33. if ~isempty(fig)
34. saveas(fig, 'Diff\_T.fig');
35. saveas(fig, 'Diff\_T.png');
36. end
37. fig = plot\_all\_thermometers(H, "Diff\_1");
38. if ~isempty(fig)
39. saveas(fig, 'Diff\_RH.fig');
40. saveas(fig, 'Diff\_RH.png');
41. end
42. end
43. end
44. function varargout = create\_table\_thermometer(varargin)
45. %% Funkcia create\_table\_thermometer.:
46. % Vstupné argumenty.:
47. % PATH = Cesta k priečinku, v ktorom sa nachádza aspoň jeden taký
48. % pod-priečinok, kde sú uložené údajové súbory s \*.edf príponou.
49. PATH = varargin{1};
50. % Definujú sa sériové čísla teplomerov.:
51. ID = [
52. "MyAmbience\_7EF0";
53. "MyAmbience\_4F95";
54. "MyAmbience\_AD3D";
55. "MyAmbience\_DC93";
56. "MyAmbience\_D577";
57. "MyAmbience\_703C";
58. "MyAmbience\_AEE2";
59. "MyAmbience\_CC2C"
60. ];
61. ID = table((0:length(ID) - 1)', ID,...
62. 'VariableNames', {'Key', 'Value'});
63. % Definujú sa typy meranej veličiny teplomerov.:
64. TYPE = [
65. "HUMIDITY";
66. "TEMPERATURE"
67. ];
68. TYPE = table((0:length(TYPE) - 1)', TYPE,...
69. 'VariableNames', {'Key', 'Value'});
70. % Extrahuje cesty k súborom a množstvo súborov.:
71. DS = datastore(PATH,...
72. 'Type', "tabulartext",...
73. 'FileExtensions', ".edf",...
74. 'ReadVariableNames', false);
75. FILES = DS.Files;
76. L = size(FILES, 1);
77. fprintf("V priečinku sa našlo %d platných súborov.\n", L);
78. fprintf("Príprava tabuľky...\n");
79. % Pre-alokácia tabuľky s názvom senzor.:
80. sensor = table;
81. sensor.ID = ID.Value;
82. sensor.HUMIDITY{size(ID, 1)} = timetable;
83. sensor.TEMPERATURE{size(ID, 1)} = timetable;
84. % Naplnenie tabuľky senzor údajmi z \*.edf súborov.:
85. for i = 0:size(ID, 1) - 1
86. for j = 0:size(TYPE, 1) - 1
87. ID\_V = ID.Value(ID.Key == i);
88. TYPE\_V = TYPE.Value(TYPE.Key == j);
89. COND\_1 = contains(FILES, ID\_V);
90. COND\_2 = contains(FILES, TYPE\_V);
91. FILE = FILES(COND\_1 & COND\_2);
92. NAME = (ID\_V + "\_" + TYPE\_V);
93. data = prepare\_thermometer(FILE, NAME);
94. sensor.(j + 2){i + 1, 1} = data;
95. end
96. end
97. fprintf("Tabuľka bola úspešne pripravená.\n");
98. % Výstupné argumenty.:
99. % varargout{1} = Tabuľka s údajmi meranej vlhkosti vzduchu.
100. % varargout{2} = Tabuľka s údajmi meranej teploty.
101. varargout{1} = table(sensor.ID, sensor.HUMIDITY,...
102. 'VariableNames', {'ID', 'Data'});
103. varargout{2} = table(sensor.ID, sensor.TEMPERATURE,...
104. 'VariableNames', {'ID', 'Data'});
105. end
106. function varargout = prepare\_thermometer(varargin)
107. %% Funkcia prepare\_thermometer.:
108. % Vstupné argumenty funkcie.:
109. % FILE\_PATH = Cesta k \*.edf súboru.
110. % FILE\_NAME = Názov výstupného \*.csv súboru.
111. FILE\_PATH = varargin{1};
112. FILE\_NAME = varargin{2};
113. try
114. % Načíta údaje.:
115. tab = readtable(string(FILE\_PATH),...
116. 'ReadVariableNames', false,...
117. 'FileType', "text");
118. % ETL proces prípravy údajov.:
119. tab(1:2, :) = [];
120. tab.Var1 = replace(tab.Var1, ",", ".");
121. tab.Var1 = str2double(tab.Var1);
122. tab.Var1 = floor(tab.Var1);
123. date = datestr(tab.Var1 / 86400 + datenum(1970, 1, 1),...
124. 'dd-mmm-yyyy HH:MM:SS');
125. tab.Var1 = datetime(date,...
126. 'InputFormat', 'dd-MMM-yyyy HH:mm:ss',...
127. 'Format', 'dd-MMM-yyyy HH:mm:ss');
128. tab = timetable(tab.Var1, tab.Var2);
129. tab = retime(tab, unique(tab.Time));
130. tab.Properties.VariableNames = {'Value'};
131. tab.Value = replace(tab.Value, ",", ".");
132. tab.Value = str2double(tab.Value);
133. tab = rmmissing(tab);
134. tab.Diff\_1 = [0; diff(tab.Time)];
135. tab.Diff\_2 = [-1; diff(tab.Diff\_1)];
136. tab.Good = tab.Diff\_2 ~= 0;
137. tab.CUMSUM = cumsum(tab.Good);
138. % Uloží tabuľku do \*.csv súboru pod vstupným názvom.:
139. writetimetable(tab, FILE\_NAME + ".csv");
140. % Výstupné argumenty.:
141. % varargout{1} = Finálna tabuľka údajov.:
142. varargout{1} = tab;
143. catch
144. % Výstupné argumenty.:
145. % varargout{1} = Prázdna tabuľka údajov.
146. tab = [];
147. varargout{1} = tab;
148. end
149. end
150. function varargout = plot\_all\_thermometers(varargin)
151. %% Funkcia plot\_all\_thermometers.:
152. % Vstupné argumenty.:
153. % DATA = Typ meraných údajov. (Napríklad T pre teplotu, H pre vlhkosť)
154. % COL = Stĺpec údajov. (Napríklad "Diff\_1" pre prvú diferenciu časového
155. % vektora)
156. DATA = varargin{1}.Data;
157. names = varargin{1}.ID;
158. COL = varargin{2};
159. L = length(DATA);
160. % Paleta farieb definovaná pre každú krivku (krivka = sériové číslo
161. % teplomera).:
162. colors = [
163. 0.000, 0.447, 0.741;
164. 0.850, 0.325, 0.098;
165. 0.929, 0.694, 0.125;
166. 0.494, 0.184, 0.556;
167. 0.466, 0.674, 0.188;
168. 0.301, 0.745, 0.933;
169. 0.635, 0.078, 0.184;
170. 0.250, 0.250, 0.250
171. ];
172. % Vytvorí prázdne plátno ako premennú.:
173. varargout{1} = figure(ID, 'Visible', 'off');
174. % Vytvorí vektor dostupných kriviek.:
175. index = [];
176. for i = 1:L
177. if ~isempty(DATA{i})
178. index = [index, i]; %#ok<AGROW>
179. end
180. end
181. % Vykreslí dostupné krivky.:
182. if ~isempty(index)
183. first = index(1);
184. plot(DATA{first}.Time, DATA{first}{:, COL});
185. varargout{1}.Color = colors(first, :);
186. if length(index) > 1
187. hold('on');
188. for i = index(2:end)
189. plot(DATA{i}.Time, DATA{i}{:, COL}, 'Color', colors(i, :));
190. end
191. end
192. labels = names(index);
193. labels = replace(labels, "MyAmbience\_", "");
194. title(COL);
195. legend(labels, 'Location', 'best');
196. hold('off');
197. % Výstupné argumenty.:
198. % varargout{1} = Grafický objekt s grafom.
199. else
200. % Výstupné argumenty.:
201. % varargout{1} = Prázdny grafický objekt.
202. varargout{1} = [];
203. end
204. end

**16. Kód skriptu *create\_voltage\_data***

1. % Zatvorí všetky otvorené okna programu MATLAB, vymaže všetky premenné a
2. % obsah príkazového riadku.:
3. close all; clear; clc
4. % Definuje plné cesty k priečinkom, s ktorými funkcia potrebuje pracovať.:
5. SCRIPT\_PATH = split(string(mfilename('fullpath')), "\");
6. SCRIPT\_PATH = join(SCRIPT\_PATH(1:end - 1), "\");
7. BASE\_PATH = split(SCRIPT\_PATH, "\");
8. BASE\_PATH = join(BASE\_PATH(1:end - 2), "\");
9. PREPARED\_PATH = BASE\_PATH + "\2. Pripravené údaje\";
10. % Volanie pomocnej funkcie join\_data pre zlúčenie jednotlivých údajových
11. % súborov.:
12. a = join\_data("a");
13. d = join\_data("d");
14. e = join\_data("e");
15. ff = join\_data("ff0", "ff1", "ff2");
16. gg = join\_data("gg0", "gg1", "gg2");
17. hh = join\_data("hh");
18. ii = join\_data("ii");
19. jj = join\_data("jj");
20. kk = join\_data("kk", "kk0new");
21. ll = join\_data("ll0", "ll1new", "ll2");
22. mmm = join\_data("mmm0", "mmm1", "mmm2");
23. nn = join\_data("nn0", "nn1", "nn2");
24. oo = join\_data("oo0", "oo1", "oo2");
25. pp = join\_data("pp0");
26. qq = join\_data("qq0", "qq1");
27. rr = join\_data("rr0");
28. ss = join\_data("ss0", "ss1", "ss2");
29. tt = join\_data("tt0", "tt1", "tt2");
30. uu = join\_data("uu0", "uu1", "uu2");
31. vv = join\_data("vv0", "vv1asi", "vv2");
32. ww = join\_data("ww0", "ww1asi", "ww2");
33. xx = join\_data("xx0bezzac", "xx1", "xx2", "xx3");
34. yy = join\_data("yy0", "yy1", "yy2");
35. % Uloží všetky tabuľky do priečinku pripravených údajov do súboru
36. % voltage\_data.mat.:
37. save(PREPARED\_PATH + 'voltage\_data.mat', '-v7.3');
38. %
39. % Zrejme chybné merania... Je potrebne ich samostatne prešetriť!
40. % o01 = join\_data("o01a", "o01b");
41. % o02 = join\_data("o02a", "o02b", "o02c", "o02d");
42. % zz = join\_data("zz0", "zz1", "zz2");
43. %
44. % Pomocná funkcia na zlúčenie viacerých tabuliek v mieste ich priesečníka.:
45. function varargout = join\_data(varargin)
46. % Definuje plné cesty k priečinkom, s ktorými funkcia potrebuje pracovať.:
47. SCRIPT\_PATH = split(string(mfilename('fullpath')), "\");
48. SCRIPT\_PATH = join(SCRIPT\_PATH(1:end - 1), "\");
49. BASE\_PATH = split(SCRIPT\_PATH, "\");
50. BASE\_PATH = join(BASE\_PATH(1:end - 2), "\");
51. DATA\_PATH = BASE\_PATH + "\1. Pôvodné údaje\Meranie U na skrate\"; %#ok<NASGU>
52. PREPARED\_PATH = BASE\_PATH + "\2. Pripravené údaje\";
53. % Cyklus, v ktorom prebieha spájanie jednotlivých vstupných tabuliek.
54. % Funkcia intersect vyhľadá priesečník medzi údajmi a následne ich zlúči do
55. % jednej tabuľky s názvom t1.:
56. for i = 1:nargin
57. eval("t" + i + "\_name = varargin{" + i + "};");
58. eval("t" + i + " = readtable(DATA\_PATH + t" + i + "\_name + '.csv');");
59. eval("t" + i + "(:, 'Var1') = [];");
60. if i >= 2
61. eval("is = intersect(t1, t" + i + ", 'stable');");
62. idx = ismember(t1, is);
63. t1(idx, :) = []; %#ok<AGROW>
64. eval("t1 = [t1; t" + i + "];");
65. end
66. end
67. % Uloží obrázok priebehu vo formáte \*.png do zložky pripravených údajov.:
68. f = figure('Visible', 'off');
69. plot(t1.Var2);
70. ylabel("Odpor vodiča (\mu\Omega)");
71. saveas(f, PREPARED\_PATH + t1\_name.extractBetween(1, 1) + '.png');
72. % Funkcia vráti finálnu tabuľku ako výstupný parameter.:
73. varargout{1} = t1;
74. end