# Pseudokody dla funkcji testALL i simulateExponsureTEST

Dokument zawiera szczegółowe pseudokody dla dwóch funkcji:  
1. `testALL` - zarządza uruchamianiem i organizacją symulacji.  
2. `simulateExponsureTEST` - wykonuje pojedynczą symulację ekspozycji z równoległym przetwarzaniem danych.

## 1. Pseudokod funkcji testALL

Pseudokod - testALL - Obsługa Równoległej Symulacji

FUNKCJA testALL(choice):

// Resetowanie puli wątków

RESETUJ pule wątków głównych (liczba\_dzialajacych\_watkow)

RESETUJ pule wątków dla operacji plikowych (4 wątki)

INICJALIZUJ lista aktywnych ubezpieczycieli:

DLA KAZDEGO w zakresie ubezp\_nazwy:

JEŻELI flagi != 0 TO:

DODAJ ubezp\_nazwy DO testVec

WYPISZ nazwy aktywnych ubezpieczycieli

USTAW `fileNames` jako lista ubezpieczycieli z `testVec`

JEŻELI choice == 1 TO:

// \*\*ŁADOWANIE DANYCH WEJŚCIOWYCH\*\* //

INICJALIZUJ struktury danych wejściowych:

DLA wojewodztwo OD 0 DO 16:

ZMIENNY wojewodztwo posiada wektory 12 miesięcy dla:

exponsure\_longitude, exponsure\_latitude, exponsure\_insurance, exponsure\_reassurance, exponsure\_sum\_value

OKREŚL ścieżkę `dane\_wejsciowe`

OKREŚL, czy aktywować `odnowienia`

OKREŚL `year` jako wybrany rok symulacji

WYPISZ listę ubezpieczycieli do przetwarzania

ROZPOCZNIJ mierzenie czasu ładowania danych

INICJALIZUJ licznik wierszy `count\_rows` = 0

LICZ ILOŚĆ WIERSZY W PLIKACH CSV:

count\_rows += LICZ\_WIERSZE(dane\_wejsciowe + "/Parametryzacja/Reasekuracja/", fileNames) \* 2

count\_rows += LICZ\_WIERSZE(dane\_wejsciowe, "/Ubezpieczyciele/", fileNames, year, odnowienia)

count\_rows += LICZ\_WIERSZE(dane\_wejsciowe + "/Parametryzacja/Pr\_pozaru.csv")

count\_rows += LICZ\_WIERSZE(dane\_wejsciowe + "/Parametryzacja/pr\_rozprzestrzenienia.csv")

count\_rows += LICZ\_WIERSZE(dane\_wejsciowe + "/Parametryzacja/pr\_wielkosc\_pozaru.csv")

PRZETWARZANIE PLIKÓW CSV:

WYKONAJ `processReas`

WYKONAJ `processOblig`

WYKONAJ `processBudynki`

WYKONAJ `processPrPozaru`

WYKONAJ `processPrRozprzestrzenienia`

WYKONAJ `processPrWielkoscPozaru`

USTAW `pasek\_postepu\_wczytywania\_danych = 1.0`

OBLICZ CZAS ładowania danych i WYPISZ

JEŻELI choice == 2 TO:

// \*\*PRZYGOTOWANIE DO SYMULACJI\*\* //

PRZETWÓRZ wartości katastroficznych szkód i minimalnych szkód:

DLA KAZDEGO elementu W `wartosc\_katastrof\_szkody\_kilka\_ubezp`:

PRZEKONWERTUJ do wartości liczbowej

JEŻELI BŁĄD KONWERSJI TO:

WYPISZ komunikat błędu

USTAW liczbę symulacji `sim = liczba\_symulacji`

USTAW licznik `licznik\_sym = sim \* 2`

ROZPOCZNIJ pomiar czasu symulacji

UTWÓRZ katalog `nazwakatalogu = createFolder(gdzie\_zapisac)`

// \*\*URUCHOMIENIE WĄTKÓW ZAPISU\*\* //

DLA OD 0 DO liczba\_watkow\_do\_zapisu:

URUCHOM `watekZapisPierwotny` W ODRĘBNYM WĄTKU

URUCHOM `watekZapisRozprz` W ODRĘBNYM WĄTKU

// \*\*URUCHOMIENIE SYMULACJI RÓWNOLEGŁEJ\*\* //

DLA sim\_num OD 0 DO sim:

URUCHOM `simulateExponsureTEST` w ODRĘBNYM WĄTKU za pomocą `pool.detach\_task(...)`

CZEKAJ NA ZAKOŃCZENIE WSZYSTKICH WĄTKÓW `pool.wait()`

OBLICZ CZAS TRWANIA SYMULACJI I WYPISZ

// \*\*ZAPISYWANIE WYNIKÓW SYMULACJI\*\* //

UTWÓRZ `Index.csv`

ZAPISZ indeksy ubezpieczycieli

// \*\*OBLICZ LICZBĘ SZKÓD DO ZAPISU\*\* //

USTAW `count\_szkody\_zapis = 0`

DLA KAŻDEGO ubezpieczyciela:

DODAJ ROZMIAR `out\_brutto\_final` DO `count\_szkody\_zapis`

OBLICZ `szkody\_zapis\_step = 1.0 / count\_szkody\_zapis`

// \*\*ZAPISYWANIE DANYCH DO PLIKÓW CSV\*\* //

DLA KAZDEGO `ubezpieczyciela`:

WYWOŁAJ `zapiszDoCSV` dla `out\_brutto\_final`, `out\_brutto\_kat\_final`, `out\_netto\_final`, `out\_netto\_kat\_final`

// \*\*ZORGANIZOWANIE WYNIKÓW W FOLDERACH\*\* //

DLA KAZDEGO `ubezpieczyciela`:

UTWÓRZ katalogi `Pierwotne` i `Rozprzestrzeniony`

ZAPISZ dane w podfolderach `Brutto`, `Netto`, `Brutto\_Kat`, `Netto\_Kat`

// \*\*CZYSZCZENIE PAMIĘCI\*\* //

WYCZYŚĆ `out\_brutto\_final`, `out\_brutto\_kat\_final`, `out\_netto\_final`, `out\_netto\_kat\_final`

WYPISZ "Symulacje zostały zapisane."

## 2. Pseudokod funkcji simulateExponsureTEST

Pseudokod dla funkcji simulateExponsureTEST

FUNKCJA simulateExponsureTEST(nazwa\_katalogu, sim, numer\_symulacji, kat\_val, ilosc\_ubezpieczycieli, num\_watku):

USTAW `krok\_symulacji = 1.0 / (17 \* 12 + ilosc\_ubezpieczycieli)`

USTAW `krok\_paska\_postepu = (1.0 / ((17 + ilosc\_ubezpieczycieli) \* sim))`

USTAW `rozmiar\_wartosci\_katastrof = ROZMIAR(wartosci\_katastrof\_szk)`

USTAW `pasek\_postepu[num\_watku] = 0.0`

INICJALIZUJ zmienne dla wartości strat, reasekuracji, indeksów

INICJALIZUJ struktury wynikowe:

symulacja\_brutto, symulacja\_brutto\_kat, symulacja\_netto, symulacja\_netto\_kat

pozary\_pierwotne, pozary\_rozprzestrzenione

USTAW indeks\_tabeli = 0

// \*\*PRZECHODZENIE PRZEZ WOJEWÓDZTWA I MIESIĄCE\*\* //

DLA wojewodztwo OD 0 DO 16:

DLA miesiac OD 0 DO 11:

POBIERZ liczba\_ekspozycji = LICZBA budynków w danym regionie i miesiącu

JEŻELI liczba\_ekspozycji > 0 TO:

WYLOSUJ liczba\_pozarow = LICZBA pożarów pierwotnych w danym regionie i miesiącu

JEŻELI liczba\_pozarow > 0 TO:

WYLOSUJ lista\_pozarow (losowy wybór budynków objętych pożarem)

DLA KAZDEGO budynek W lista\_pozarow:

ZWIĘKSZ indeks\_tabeli O 1

POBIERZ numer\_budynku

// \*\*WYSZUKIWANIE BUDYNKÓW W PROMIENIU 200m\*\* //

OBLICZ budynki\_w\_poblizu = WYSZUKAJ\_BUDYNKI\_W\_PROMIENIU(200, numer\_budynku, wojewodztwo, miesiac)

// \*\*SPRAWDZENIE, NA KTÓRE BUDYNKI POŻAR SIĘ ROZPRZESTRZENI\*\* //

OBLICZ rozprzestrzenienie\_pozaru NA PODSTAWIE budynki\_w\_poblizu

OBLICZ procent\_straty\_pozarowej ORAZ wartosc\_straty\_brutto

JEŻELI wartosc\_straty\_brutto < 500 TO:

USTAW wartosc\_straty\_brutto = 500

POBIERZ ubezpieczyciel budynku

OBLICZ suma\_reasekuracji

// \*\*DODANIE POŻARU PIERWOTNEGO DO BUFORA\*\* //

PRZENIEŚ `pozary\_pierwotne` DO `global\_buffer\_pierwotny` WĄTKOWO

OBLICZ liczba\_budynkow\_rozprzestrzenionych

JEŻELI liczba\_budynkow\_rozprzestrzenionych > 0 TO:

// \*\*OBLICZENIA KATASTROFICZNE I REASEKURACYJNE\*\* //

OBLICZ wartosci\_katastroficzne = OBLICZ\_WARTOSCI\_KATASTROFICZNE(

dane\_pozary\_pierwotne, dane\_pozary\_rozprzestrzenione, kat\_val, ilosc\_ubezpieczycieli, ubezpieczyciel, wartosc\_straty\_brutto)

// \*\*GRUPOWANIE STRAT MIĘDZY UBEZPIECZYCIELI\*\* //

PRZYDZIEL szkody pomiędzy ubezpieczycieli na podstawie wartosci\_katastroficznych

OBLICZ dane\_symulacyjne\_brutto

OBLICZ ewentualne straty katastroficzne

OBLICZ ryzyko\_reasekuracyjne

ZAPISZ wyniki w wektorach symulacyjnych

// \*\*DODANIE POŻARÓW ROZPRZESTRZENIONYCH DO BUFORA\*\* //

PRZENIEŚ `pozary\_rozprzestrzenione` DO `global\_buffer\_rozprz` WĄTKOWO

W PRZECIWNYM RAZIE:

ZAPISZ pożar w podstawowych wektorach symulacyjnych

SPRAWDŹ czy straty katastroficzne muszą zostać zapisane

AKTUALIZUJ pasek\_postepu oraz stan\_symulacji

// \*\*ZAPIS DANYCH DO BUFORA I PLIKÓW CSV\*\* //

JEŻELI forma\_zapisu\_budynkow == 0 TO:

PRZENIEŚ pozary\_pierwotne DO `global\_buffer\_pierwotny`

PRZENIEŚ pozary\_rozprzestrzenione DO `global\_buffer\_rozprz`

// \*\*PRZETWARZANIE DANYCH Z BUFORÓW\*\* //

DLA KAZDEGO `bufor\_pierwotny` W `global\_buffer\_pierwotny`:

PRZETWARZAJ `bufor\_pierwotny` WĄTKOWO

DLA KAZDEGO `bufor\_rozprzestrzeniony` W `global\_buffer\_rozprz`:

PRZETWARZAJ `bufor\_rozprzestrzeniony` WĄTKOWO

// \*\*AGREGACJA WYNIKÓW I FINALNE ZAPISY\*\* //

ZABLOKUJ zasoby i agreguj dane dla każdego ubezpieczyciela

DLA KAZDEGO ubezpieczyciela OD 0 DO ilosc\_ubezpieczycieli:

OBLICZ suma\_brutto na podstawie symulacja\_brutto

WYCZYŚĆ symulacja\_brutto

ZAPISZ suma\_brutto do wyników

OBLICZ suma\_brutto\_kat na podstawie symulacja\_brutto\_kat

WYCZYŚĆ symulacja\_brutto\_kat

ZAPISZ suma\_brutto\_kat do wyników

OBLICZ suma\_netto na podstawie symulacja\_netto

WYCZYŚĆ symulacja\_netto

ZAPISZ suma\_netto do wyników

OBLICZ suma\_netto\_kat na podstawie symulacja\_netto\_kat

WYCZYŚĆ symulacja\_netto\_kat

ZAPISZ suma\_netto\_kat do wyników

AKTUALIZUJ pasek\_postepu oraz stan\_symulacji