Zadania według profilu A: to zadania, w których w elementach macierzy przykładowo: 3 na 3 elementy umieszczamy wartości numerycznie wyznaczanych całek oznaczonych w granicach:
- ***a*** dolna granica – niezmiennie równa zeru, bądź bezwzględnej różnicy wskaźnika kolumny i wiersza,
- ***b*** górna granica – równa na przykład sumie, bądź iloczynowi wskaźnika kolumny i wiersza, każdego z elementów tej macierzy.

Innymi słowy reguły określania dolnej i górnej granicy numerycznego całkowania powinny być określone, jako pochodna wskaźnika kolumny i wskaźnika wiersza, każdego z na bieżąco wyznaczanego elementu tej macierzy. Stąd istnieje możliwość stosowania pewnej dowolności w definicji zadania profilu A, prócz tych 11 przykładów umieszczonych poniżej.

1. Wyznaczyć wartości macierzy 3 na 3 elementów, o wartościach całek oznaczonych z sinusa (‘**sin**’) w granicach: dolna - niezmiennie równa zero z elementu na element tej macierzy, natomiast górna granica równa sumie wskaźnika bieżącego wiersza i kolumny każdego wyznaczanego elementu.

2. Wyznaczyć wartości macierzy 4 na 4 elementy, o wartościach całek oznaczonych z kosinusa (‘**cos**’) w granicach: dolna – równa wartości bezwzględnej wskaźnika bieżącego wiersza i kolumny elementu, natomiast górna granica równa sumie wskaźnika bieżącego wiersza i kolumny elementu wyznaczanego.

3. Wyznaczyć wartości macierzy 2 na 2 elementy o wartościach całek oznaczonych z funkcji gamma (‘**gamma**’) w granicach: dolna – niezmiennie równa 0.02 z elementu na element tej macierzy, natomiast górna granica równa sumie wskaźnika bieżącego wiersza i kolumny każdego wyznaczanego elementu

4. Wyznaczyć wartości macierzy 3 na 3 elementy o wartościach całek oznaczonych z funkcji arkus sinus (‘**asin**’) w graniach: dolna – równa wskaźnikowi wiersza bieżącego elementu tej macierzy, natomiast górna granica równa sumie wskaźnika bieżącego wiersza i kolumny każdego wyznaczanego elementu.

5. Wyznaczyć wartości macierzy 4 na 4 elementy o wartościach całek oznaczonych z funkcji arkus cosinus (‘**acos**’) w granicach: dolna – równa wskaźnikowi kolumny bieżącego elementu tej macierzy, natomiast górna granica równa podwojonej sumie wskaźnika bieżącego wiersza i kolumny każdego wyznaczanego elementu.

6. Wyznaczyć wartości macierzy 5 na 5 elementów o wartościach całek oznaczonych z funkcji arkus sinus hiperboliczny (‘**asinh**’) w granicach: dolna – równa różnicy bezwzględnej wskaźnika kolumny i wiersza bieżącego elementu tej macierzy, natomiast górna granica równa potrojonej sumie wskaźnika bieżącego wiersza i kolumny każdego wyznaczanego elementu.

7. Wyznaczyć wartości macierzy 4 na 4 elementy o wartościach całek oznaczonych z funkcji arkus kosinus hiperboliczny (‘**acosh’**) w granicach: dolna – równa różnicy bezwzględnej wskaźnika kolumny i wiersza bieżącego elementu tej macierzy, natomiast górna granica równa podwójnej sumie wskaźnika bieżącego wiersza i kolumny każdego wyznaczanego elementu.

8. Wyznaczyć wartości macierzy 5 na 5 elementów o wartościach całek oznaczonych z funkcji logarytm naturalny (‘**log**’) w granicach: dolna – równa zeru niezmiennie z elementu na element tej macierzy, górna równa podwojonej sumie wskaźnika kolumny i wiersza każdego z elementów.

9. Wyznaczyć wartości macierzy 6 na 6 elementów o wartościach całek oznaczonych z funkcji eksponent (‘**exp**’) w granicach: dolna – równa bezwzględnej różnicy wskaźnika kolumny i wiersza każdego z elementów tej macierzy, górna równa iloczynowi wskaźnika kolumny i wiersza każdego z elementów tej macierzy. *UWAGA: w Octavii być może koniecznym będzie zrzutowanie typu float iloczynu kolumny i wiersza do typu krótkiego całkowito liczbowego, czyli zamiast stosowania iloczynu:* **i*j**, *winno być: uint8(i*j)*

10. Wyznaczyć wartości macierzy 3 na 3 elementy o wartościach całek oznaczonych z funkcji logarytm dziesiętny (‘**log10**’) w granicach: dolna – niezmiennie równa zeru z element na element wyznaczany tej macierzy, natomiast górna granica równa poczwórnej sumie kolumny i wiersza każdego na bieżąco wyznaczanego elementu.

11. Wyznaczyć wartości macierzy 5 n 5 elementów o wartościach całek oznaczonych z funkcji logarytm przy podstawie dwa (‘**log2**’) w granicach: dolna – równa bezwzględnej różnicy wskaźnika kolumny i wiersza każdego, z na bieżąco wyznaczanego elementu tej macierzy, natomiast górna równa podwojonej sumie, bądź iloczynowi wskaźnika kolumny i wiersza każdego z elementów tego wiersza.

Zadania według profilu B to zadania, w których w elementach macierzy przykładowo: 3 na 3 elementy umieszczamy wartości rekurencyjnie wyznaczanych elementów szeregu Fibonacciego, bądź takiego ciągu hiper-Fibonacciego (zaczynającego się od trzech jedynek, i określającego element następny, jako sumę trzech ostatnich elementów tego szeregu), czyli szeregu trójkowego, bądź ciągu czwórkowego (zaczynającego się elementami {1,1,1,1}, a o postaci następnego elementu w tym ciągu czwórkowym wyznaczanym, jako sumę 4 ostatnich poprzednich elementów tego szeregu).

Innymi słowy reguły określania elementów macierzy na przykład 3 na 3, lub 4 na 4 elementy dotyczą w istocie podwywołania funkcji rekurencyjnego wyznaczania elementów ciągu Fibonacciego, bądź ciągu trójkowego, bądź ciągu czwórkowego, omawianych na Zajęciach. Jednakże w ogólności mogą również dotyczyć wyznaczania elementów dowolnego ciągu liczb naturalnych podobnych w naturze do wspomnianego ciągu Fibonacciego, według wzoru kilku konkretnych treści przedstawionych poniżej:

1. Wyznaczyć wartości macierzy 3 na 3 elementów, o wartościach ciągu Fibonacciego, o numerach porządkowych będących sumą wskaźnika kolumny i wiersza każdego z na bieżąco wyznaczanych elementów tej macierzy.

2. Wyznaczyć wartości macierzy 4 na 4 elementy, o wartościach ciągu trójkowego, o numerach porządkowych będących bezwzględną różnicą kolumny i wiersza każdego z na bieżąco wyznaczanych elementów tej macierzy

3. Wyznaczyć wartości macierzy 2 na 2 elementy, o wartościach ciągu trójkowego, o numerach porządkowych będących podwojoną sumą kolumny i wiersza każdego z na bieżąco wyznaczanego elementu tej macierzy.

4. Wyznaczyć wartości macierzy 4 na 4 elementy, o wartościach ciągu trójkowego, o numerach porządkowych będących iloczynem kolumny i wiersza każdego z na bieżąco wyznaczanego elementu tej macierzy.

5. Wyznaczyć wartości macierzy 3 na 3 elementy, o wartościach ciągu Fibonacciego, o numerach porządkowych będących sumą kolumny i wiersza każdego z na bieżąco wyznaczanego elementu tej macierzy, jednakże o wartościach elementów tej macierzy, powiększonych o wartości kwadratu chińskiego 4 n 4 elementy.

Poniżej przytoczono w ramce funkcję nadrzędną wywołań:

```
%% Przykładowe zadanie wykładowe zwracające macierz 'wymiar' na 'wymiar' elementów,%
%% będących wyznaczoną funkcją z argumentu: sumy kolumny i wiersza  bieżącego tej macierzy%
%% Przykładowe wywołanie funkci zadaniewyk:
%%-----------------------------------------
% [A] = zadaniewyk(3,'sin',0.01); %<= policz macierz 3x3 elem.,będących sinusem z arg. sumy wiersz i kolumny w tej mac%
function [A] = zadaniewyk(wymiar,fn,dokl)
A = zeros(wymiar,wymiar);
for i = 1 : wymiar,
    for j = 1 : wymiar,
          %A(i,j) = feval(fn,i+j);
          A(i,j) = dowolnafunkcjadokl(0,i+j,dokl,fn);%WARIANT A ZADAŃ
          %A(i,j) = fibcaserek(i+j);  %wariant B zadań
    end;
end;
```

Poniżej przytoczono funkcję I rzędu podrzędną całkującą z przymierzaniem trapezów, z zadanym bezwzględnym marginesem błędów wyniku ostatniego i przedostatniego:

```
%Przykład wywołania całkowania ze skryptem nadrzędnym z użyciem dowolnej prostej(jednokrotnej) funkcji:%
%-----------------------------------------------------------------------------%
% pole=dowolnafunkcjadokl(0,1,0.001,'sin') %<=wyznacz pole - całk. oznaczoną pod funkcją sinus w granicach [0,1radian]%
% pole=dowolnafunkcjadokl(0,1,0.001,'cos') %<=wyznacz pole - całk. oznaczoną pod funkcją sinus w granicach [0,1radian]%
% pole=dowolnafunkcjadokl(0,1,0.001,'exp') %<=wyznacz pole - całk. oznaczoną pod funkcją eksponent(x) w granicach
[0,1radian]%
function [pole] = dowolnafunkcjadokl(a,b,dkl,fn) %przykładowo z wywołaniem: dowolnafunkcjadokl(0,1,0.001,'sin');%
pole = 0;    % pole inicjuj na zerowe
n = 5;       % zacznij od 5 podprzedziałów
rozn_wynik = 10;  % zmienna określ. nieco dalej różnicę bezwzgl. wyników: ostatniego z przedostatnim%
szr = abs(b-a)/n; % wyznacz szerokość każdego z podprzedziałów
%figure,          % przygotuj (tzn. otwórz) okno wydruku graficznego danych
while(dkl < rozn_wynik)  % dopóki szacowany błąd nie jest mniejszy niż deklarowany margines błędu%
    %OsX = a:szr:b;       % wyznacz WEKTOR argumentu całego odcinka osi X w przedz.[a,b] %
    %Wielomian = feval(fn,OsX); % wyznacz WEKTOROWO przeciw-argument osi X (wartość funkcji)
    %plot(OsX,Wielomian,'r');% wstępnie na czerwono wyświetl linią ciągłą funkcję
    %hold on;                % przytrzymaj uchwyt okna wyświetleń graficznych danych
    %bar(OsX,Wielomian,0.975,'b'); % dorysuj na wykr. 'dość zgrubnie' szereg słupków-prost., przymierz.do krzywej%
    %disp('Proszę wprowadzić: ''return'' albo sprawdzić np. wartości zmiennych roboczych :');
    %keyboard;               % przekaż sterowanie do do promptu użytkownika: K>>_
    %hold off;               % zwolnij uchwyt okna graficznego, by w nast. pętli odświeżyć wykres
    pole_przedostatnie = dowolnafunkcja(a,b,n,fn); % policz pole - całki oznaczonej zgrubnie
    n=2*n                              % zdubluj liczbę podprzedziałów
    szr=szr/2;                         % zmniejsz odpowiednio ich szerokości
    pole_ostatnie      = dowolnafunkcja(a,b,n,fn); % a obecnie: policz pole - całki oznaczonej dokładniej!
    rozn_wynik         = abs(pole_ostatnie - pole_przedostatnie); % szacuj margines błędu
end;
pole = pole_ostatnie; % de facto: pole ostatnie, to te najdokładniejsze!
```

Poniżej przytoczono funkcję 2 rzędu podrzędną całkującą z przymierzaniem trapezów, z zadaną liczbą podprzedziałów, wywoływaną jednakże przez funkcję przedstawioną powyżej I rzędu (z określeniem marginesu bezwzględnego błędu):

```
%Przykład wywołania całkowania z użyciem dowolnej prostej(jednokrotnej) funkcji:%
%----------------------------------------------------------------------------%
% pole=dowolnafunkcja(0,1,10000,'sin') %<=wyznacz pole - całk. oznaczoną pod funkcją sinus w granicach
[0,1radian]
function [pole] = dowolnafunkcja(a,b,n,fn)  % wersja przybliżania trapezami pola, w reprezent. całki ozn.
pole = 0;          % inicjuj wartość pola całki oznaczonej y=x^2+x
szr = abs(b-a)/n; % wyznacz szer. każdego z podprzedz., dzieląc długość przedziału [a,b] na n pododcinków
XL=a;              % przypisz wart. lewej granicy podprzedziału - XL - lewą granicę przedziału:[a,b]
XP=a;              % przypisz wart. prawej granicy podprzedziału - XP - prawą granicę przedziału [a,b]
XSR=a;            % srodek każdego z podprzedziału - na razie domyślnie ustawiony na wartość a
for i=1:n          % w pętli n-krotnej dosumowywuj pola prostokątów podprzedziałów o podstawie: 'szr'
    XP = XP + szr;  % przed właściwym doliczeniem pola trapezu - ustaw wart. prawej granicy podprzedziału
    XSR = (XL+XP)/2;% określ położenia środka podstawy trapezu (na osi argumentu X)
    pole = pole + szr*abs(feval(fn,XSR)); %dosumuj w pętli pole jednego trapezu w podprzedz.[XL,XP]
    XL = XP + szr;  % przed końcem każdego przebiegu pętli 'for' przesuń XP o szer. 1 podprzedz.
end;
```